

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

ÉTUDE COMPARATIVE DES PROTOCOLES ZIGBEE PRO ET ZIGBEE IP

MÉMOIRE

PRÉSENTÉ

COMME EXIGENCE PARTIELLE

DE LA MAÎTRISE EN GÉNIE ÉLECTRIQUE

PAR

IMANE BERDAI

OCTOBRE 2016

UNIVERSITÉ DU QUÉBEC À MONTRÉAL
Service des bibliothèques

Avertissement

La diffusion de ce mémoire se fait dans le respect des droits de son auteur, qui a signé le formulaire *Autorisation de reproduire et de diffuser un travail de recherche de cycles supérieurs* (SDU-522 – Rév.07-2011). Cette autorisation stipule que «conformément à l'article 11 du Règlement no 8 des études de cycles supérieurs, [l'auteur] concède à l'Université du Québec à Montréal une licence non exclusive d'utilisation et de publication de la totalité ou d'une partie importante de [son] travail de recherche pour des fins pédagogiques et non commerciales. Plus précisément, [l'auteur] autorise l'Université du Québec à Montréal à reproduire, diffuser, prêter, distribuer ou vendre des copies de [son] travail de recherche à des fins non commerciales sur quelque support que ce soit, y compris l'Internet. Cette licence et cette autorisation n'entraînent pas une renonciation de [la] part [de l'auteur] à [ses] droits moraux ni à [ses] droits de propriété intellectuelle. Sauf entente contraire, [l'auteur] conserve la liberté de diffuser et de commercialiser ou non ce travail dont [il] possède un exemplaire.»

REMERCIEMENTS

Je tiens à exprimer ici mes sincères remerciements à toutes les personnes qui m'ont apporté leur soutien.

Je souhaite adresser de grands remerciements à Monsieur Abdellatif Obaid pour son encadrement, sa disponibilité et ses encouragements face aux difficultés que j'ai rencontrées pendant mon travail.

Je remercie également Monsieur Mounir Boukadoum pour ses conseils et pour le temps qu'il a bien voulu me consacrer.

Mes profonds remerciements à tous mes amis de l'UQAM pour leur aide et leur compréhension.

Toutes mes gratitudes s'adressent à mes chers parents qui m'ont soutenue depuis mon jeune âge. Merci pour tout l'amour que vous me donnez chaque jour.

Merci à ma famille et à mes amis pour leurs encouragements.

TABLE DES MATIÈRES

LISTE DES FIGURES	vii
LISTE DES TABLEAUX.....	ix
LISTE DES ACRONYMES	xi
RÉSUMÉ	xv
CHAPITRE I	
INTRODUCTION	1
1.1 Problématique	5
1.2 Objectif	5
CHAPITRE II	
ETAT DE L'ART	7
2.1 Comparaison de ZigBee et autres technologies sans fil.....	7
2.2 Travaux réalisés sur ZigBee-Pro et ZigBee-IP	10
CHAPITRE III	
TECHNOLOGIES UTILISÉES	16
3.1 Le standard ZigBee	16
3.1.1 WSN et le standard ZigBee	17
3.1.2 IEEE 802.15.4/ ZigBee.....	18
3.1.3 Architecture du standard ZigBee.....	19
3.1.4 Dispositifs du standard ZigBee.....	20
3.1.5 Topologies du réseau	21

3.2 ZigBee-Pro	22
3.3 ZigBee IP	24
3.3.1 Architecture de ZigBee IP	24
3.3.2 La technologie 6LoWPAN	25
3.3.3 Le protocole CoAP	29
3.4 Outils de simulation : Contiki OS/ Cooja	30
CHAPITRE IV	
RÉALISATION ET SIMULATION	32
4.1 Réalisation matérielle du réseau ZigBee-Pro	32
4.1.1 Le module XBee :	32
4.1.2 Configuration des nœuds du réseau	34
4.1.3 Première topologie :	34
4.1.4 Deuxième topologie	41
4.1.5 Troisième topologie	45
4.2 Réalisation logicielle du réseau ZigBee IP	49
4.2.1 Première simulation	49
4.2.2 Deuxième simulation	55
4.2.3 Exemple du protocole CoAP	58
4.3 Comparaison des résultats	61
CONCLUSION	63
BIBLIOGRAPHIE	65

LISTE DES FIGURES

Figure	Page
Figure 3.1: Schématisation du rôle d'un capteur	18
Figure 3.2: Architecture d'un WSN	18
Figure 3.3: Architecture du standard ZigBee.....	19
Figure 3.4: Dispositifs du standard ZigBee	21
Figure 3.5: Topologies du réseau ZigBee	21
Figure 3.6: Architecture du ZigBee-Pro	23
Figure 3.7: Architecture du réseau ZigBee IP	25
Figure 3.8: Architecture du protocole 6LoWPAN.....	26
Figure 3.9: taille de la trame de la norme IEEE 802.15.4.....	27
Figure 3.10: méthode de fragmentation 6LoWPAN.....	28
Figure 3.11: Comparaison entre deux modèles de couches OSI et 6LoWPAN	29
Figure 3.12 : Architecture du protocole CoAP	30
Figure 3.13 : Pile protocolaire Contiki	31
Figure 4.1 : Entrée/sortie du module XBee	33
Figure 4.2 : Configuration des nœuds XBee.....	33
Figure 4.3 : Architecture du premier réseau ZigBee.....	35
Figure 4.4 : Simulation du premier réseau avec X-CTU	35
Figure 4.5 : Délai de transmission des données (première topologie, expérience 1). ..	36
Figure 4.6 : Un extrait du programme Java pour l'application XBee.....	37
Figure 4.7 : Un extrait du programme Java pour le téléchargement des données	38
Figure 4.8 : Calcul de la tension des XBee en état de veille (topologie 1)	40

Figure 4.8 : Calcul de la tension des XBee en état de marche (topologie 1)	41
Figure 4.9 : Architecture du deuxième réseau ZigBee.....	42
Figure 4.10 : Simulation du deuxième réseau avec X-CTU	42
Figure 4.11 : Calcule de la tension des XBee en état de veille (topologie 2)	44
Figure 4.12 : Calcule de la tension des XBee en état de marche (topologie 2)	45
Figure 4.13 : Architecture du troisième réseau ZigBee	46
Figure 4.14 : Configuration du troisième réseau avec X-CTU	46
Figure 4.15 : Calcule de la tension des XBee en état de veille (topologie 3)	48
Figure 4.16 : Calcule de la tension des XBee en état de marche (topologie 3)	48
Figure 4.17 : Simulation du premier exemple de réseau ZigBee IP	51
Figure 4.18 : Connexion du nœud routeur	51
Figure 4.19: Activation du pont du nœud serveur	52
Figure 4.20 : Decouverte de voisin a partir du nœud serveur.....	52
Figure 4.21 : Valeur des capteurs des quatre nœuds du réseau.....	53
Figure 4.22 : Consommation d'énergie des nœuds du premier réseau.	54
Figure 4.23 : Simulation du deuxième exemple du réseau ZigBee IP	56
Figure 4.24 : Découverte de voisin pour la deuxième topologie	57
Figure 4.25 : Consommation d'énergie du deuxième réseau.....	57
Figure 4.26 : Simulation du réseau CoAP sur Contiki.....	58
Figure 4.27 : Ressources offerte par CoAP	59
Figure 4.28 : Interrogation du nœud capteur 2	60
Figure 4.29 : Changement d'état de la LED du nœud capteur 3.....	61

LISTE DES TABLEAUX

Tableau	Page
Tableau 2.1 : Exemples des Open sources des réseaux de capteurs sans fil.....	12
Tableau 2.2 : Comparaison des solutions pour les maisons automatisé	13
Tableau 2.3 : Implémentations 6LoWPAN existantes.....	14
Tableau 3.1 : Specification du standard ZigBee	17
Tableau 3.2 : Comparaison des en-têtes IPv6 et 6LoWPAN	27
Tableau 4.1 : Delai moyen topologie 1	38
Tableau 4.2 : Delai moyen topologie 2	43
Tableau 4.3 : Delai moyen topologie 3	47
Tableau 4.4 : Comparaison des résultats	49
Tableau 4.5 : Données captées par les nœuds du réseau.....	53
Tableau 4.6 : Données captées par le nœud 2	60
Tableau 4.7 : Comparaison des résultats entre ZigBee-Pro et ZigBee IP	62

LISTE DES ACRONYMES

WPAN Wireless Personal Area Network.

WMAN Wireless Metropolitan Area Network.

WWAN Wireless Wide Area Network.

GSM Global System for Mobile

GPRS General Packet Radio Service.

UMTS Universal Mobile Telecommunication System.

LTE Long Term Evolution.

WiMAX Worldwide Interoperability for Microwave Access

LP-WPAN Low Power-Wireless Personal Area Network.

AODV Ad hoc On Demand Distance Vector.

DSR Dynamic Source Routing.

CoAP Constrained Application Protocol.

AF Application Framework.

ZDO ZigBee device objects.

REST Representational state transfer.

SSP Security Service Provider.

APS Application Support Sub-Layer.

ZDO ZigBee Device Object.

FFD Full fonction device.

RFD Reduced fonction device.

ZC Coordinateur ZigBee .

ZR Routeur ZigBee.

RDC Radio Duty cycling.

UART Universal Asynchronous Receiver Transmitter.

UDGM Unit Disk Graph Medium.

OQPSK Offset quadrature phase-shift keying

DSSS Direct Sequence Spread Spectrum

RESUMÉ

Depuis l'apparition des Réseaux de Capteurs sans Fil (Wireless Sensor Network WSN) dans les années 1990, il y a eu des impacts sur plusieurs domaines tels que l'environnement, les systèmes biologiques, etc. (Fallis, 2013). Grâce au rôle pertinent des WSN qui observent, décident et réagissent aux changements dans les environnements dans lesquels ils sont utilisés.

Dans ce cadre, plusieurs standards et protocoles de communication ont été développés.

Le standard ZigBee (Banzi, 2011) a été conçu dans le but d'interconnecter des capteurs sans fil qui communiquent à bas débit et faible coût et coopérant entre eux d'une manière autonome via des ondes radio. ZigBee inclut la spécification complète des couches physiques et MAC de la norme IEEE802.15.4 pour les réseaux sans fil personnels (Wireless Personal Area Networks WPAN). Les autres couches du standard ont été ajoutées dépendamment de la version du réseau ZigBee utilisé. Ceci a amené à la définition des deux versions les plus utilisées : ZigBee Pro et ZigBee IP.

Dans ce mémoire, on a effectué une comparaison entre ces deux types du standard ZigBee. Nous avons effectué des simulations matérielles et logicielles qui ont montré une différence en termes de délais d'acheminement des informations et en terme de consommation d'énergie.

Mots clés : ZigBee, ZigBee-Pro, ZigBee IP, 6LoWPAN, IPv6

CHAPITRE I

INTRODUCTION

Le monde informatique a connu une grande expansion depuis que les réseaux sans fil ont vu le jour. Un réseau sans fil permet de lier des terminaux sans liaison filaire. L'existence de ce type de réseaux a permis plus de mobilité et de flexibilité dans la structuration des réseaux. Les utilisateurs peuvent se connecter facilement et peuvent se déplacer tout en gardant la connexion.

La norme IEEE802.11 (connue sous le nom WiFi) définit les réseaux locaux sans fil (WLAN pour Wireless Local Area Network) et offre des débits de transmission allant jusqu'à 54 Mbps sur une distance de plusieurs centaines de mètres (Phan, 2015). Elle est considérée comme la norme la plus utilisée actuellement pour ce type de réseaux.

La technologie WPAN (*Wireless Personal Area Network*) est un autre type de réseaux sans fil. Le standard Bluetooth lancé par Ericsson en 1994 est un tel réseau destiné au transfert des données légères avec une bande passante plus faible par rapport au WI-FI, proposant un débit théorique de 1 Mbps pour une portée maximale d'une trentaine de mètres. Bluetooth, connu aussi sous le nom IEEE 802.15.1, possède l'avantage d'être très peu gourmand en batterie; ce qui le rend particulièrement adapté à une utilisation au sein de petits périphériques.

Les réseaux métropolitains sans fil (WMAN pour *Wireless Metropolitan Area Network*) connus sous le nom de Boucle Radio Locale sont basés sur la norme IEEE 802.16. Une boucle locale radio offre un débit utile de 1 à 10 Mbit/s pour une portée

de 4 à 10 kilomètres; ce qui la destine principalement aux opérateurs de télécommunication. Le WiMAX (*Worldwide Interoperability for Microwave Access*) se considère comme la norme de réseau métropolitain sans fil la plus connue, il procure des débits de l'ordre de 70 Mbit/s sur une zone de couverture portant sur quelques dizaines de kilomètres.

Le réseau étendu sans fil (WWAN pour *Wireless Wide Area Network*) est également connu sous le nom de réseau cellulaire mobile. Il s'agit des réseaux sans fil les plus répandus puisque tous les téléphones mobiles sont connectés à un réseau étendu sans fil. Les principales technologies sont les suivantes :

- GSM (*Global System for Mobile*).
- GPRS (*General Packet Radio Service*)
- UMTS (*Universal Mobile Telecommunication System*)
- LTE (*Long Term Evolution*)

Les réseaux de capteurs sans fil (RCSF) sont un type particulier de réseaux Ad-hoc dans lesquels les nœuds sont des capteurs intelligents (Smart Sensors), de petite taille et équipés de fonctionnalités avancées telles qu'un microcontrôleur, un transducteur et/ou actionneur et un émetteur/récepteur sans fil à faible portée. Dans ce type de réseau, les nœuds capteurs échangent des informations sur l'environnement dans le but d'obtenir un aperçu sur une région contrôlée (Francomme,2008). Tous les dispositifs du réseau communiquent via les ondes radio.

Les WSN sont utilisés dans des domaines aussi variés dont on peut citer le secteur militaire, médical, industriel ou encore plus dernièrement le secteur environnemental, plus précisément dans l'agriculture où la technologie des RCSF offre un support important qui permettra la gestion précise des ressources, l'optimisation de l'irrigation et de l'utilisation des engrais, le suivi/prévention des maladies, la prédiction du moment adéquat de la récolte, etc. L'avantage principal de l'utilisation

des RCSF dans l'agriculture est l'absence de câblage ce qui réduit considérablement leur coût d'installation (Bechkit et al., 2011).

La médecine est l'un des secteurs marquants dans lequel les réseaux de capteurs jouent un rôle considérable permettant de suivre et d'analyser les faits et gestes des personnes concernées. Au moment où ces derniers (personnes âgées, handicapées) souffrent d'une perte d'autonomie, un domicile intelligent doté d'une haute surveillance permet d'améliorer leurs conditions de vie en leur offrant une assistance réactive adaptée à leurs besoins, en facilitant le travail des différents intervenants à domicile (professions de santé, auxiliaires de vie) et surtout en permettant de donner l'alerte en cas de survenue d'une situation critique (Chehaider et al, 2013).

«Les applications des WSN peuvent être classées en deux catégories selon (Phan,2015) : 1) la télésurveillance et 2) le suivi des objets mobiles. Contrairement aux applications de suivi des objets mobiles, qui demandent des mises à jour en temps réel des résultats, les applications de télésurveillance mesurent périodiquement les conditions d'environnement et peuvent envoyer leurs données selon trois modes :

- Périodique pendant un intervalle de temps prédéfini
- À la suite d'un événement spécifique : cela se produit souvent lorsque la valeur d'une mesure spécifique atteint un seuil prédéfini
- En réponse à une interrogation d'un utilisateur.»

Le standard ZigBee a été conçu autour vers l'année 1998 par des ingénieurs d'applications qui ont constaté que le WiFi et Bluetooth ne convenaient plus à certaines manipulations et qu'ils doivent penser à une technique d'auto-organisation des réseaux WSN. ZigBee est considéré comme une norme dont les composants sont autonomes et assurent une longue durée de vie de la batterie (alimentées pendant plusieurs mois par des piles classiques).

ZigBee est aussi un LP-WPAN (*Low Power – Wireless Personal Area Network*) à bas débit et à courte portée qui utilise les ondes hertziennes pour transporter des messages entre deux ou plusieurs entités réseaux. Il est caractérisé par une portée comprise entre quelques mètres et quelques centaines de mètres et un débit faible (maximum 250 kbits/s). La différence entre ZigBee et la plupart des autres réseaux locaux et personnels sans fil (WiFi, Bluetooth, etc.) se situe au niveau de l'utilisation du médium hertzien. ZigBee est optimisé pour une faible utilisation du médium partagé par tous, par exemple 0,1 % du temps (Anthony et Jérémy, 2011). Typiquement, un module ZigBee occupera le médium pendant quelques millisecondes en émission, attendra éventuellement une réponse ou un acquittement, puis se mettra en veille pendant une longue période avant l'émission suivante, qui aura lieu à un instant prédéterminé.

Les caractéristiques dont le standard ZigBee bénéficie ont inspiré les chercheurs dans le but de l'intégrer dans la nouvelle tendance Internet des objets (ou Internet of Things - IoT). Cette mode où tous les objets qui nous entourent seront connectés. On en a dénombré près de 15 milliards en 2015 (Nemri, 2015).

ZigBee-Pro et ZigBee-IP sont deux versions de ZigBee. Leur atout principal est de pouvoir s'interconnecter avec TCP/IP. ZigBee tout seul ne peut pas être relié à Internet. ZigBee-Pro (appelé aussi ZigBee Gateway) exploite une passerelle en plus des couches employées dans ZigBee. Cette passerelle fait la liaison et achemine les données vers Internet. Cette version améliorée fournit des fonctionnalités supplémentaires qui sont particulièrement utiles pour les grands réseaux. Ces fonctionnalités concernent le routage et la sécurité.

ZigBee-IP est considéré comme la première spécification basée sur IPv6 pour les réseaux de capteurs sans fil. Il place certains protocoles de la pile TCP/IP directement dans le réseau ZigBee; ce qui donne naissance à une couche d'adaptation appelée:

6LoWPAN (IPv6 sur LoWPAN) assurant la coexistence entre le standard IEEE802.15.4 et l'IPv6 ainsi que l'échange des paquets (IPv6) sur le réseau.

1.1 Problématique

Grâce aux progrès des réseaux de capteurs sans fil que beaucoup de domaines ont été améliorés. Un réseau de capteur se compose de coordinateurs et un nombre de capteurs. Chaque capteur peut se connecter à un ou plusieurs autres capteurs.

Un capteur peut exercer une des fonctions suivantes : détection, relai ou échange de données avec les réseaux externes. Un capteur de mesure qui peut seulement envoyer des données est appelé 'capteur de détection', celui pour relayer des données 'routeur', et celui pour échanger des données avec les autres réseaux 'coordonateur' (Phan, 2015). Les données captées par les dispositifs ou nœuds du réseau sont traitées et envoyées à un serveur pour arriver finalement à un réseau IP.

Dans le cas de notre travail sur ZigBee, nous visons l'exploitation des deux technologies ZigBee Pro et ZigBee IP, dans le but de profiter au mieux d'une meilleure qualité d'utilisation et de faciliter et améliorer leur intégration dans les applications modernes des réseaux. Nos capteurs sont alimentés par des batteries, dont la longévité est limitée. Contrôler cette limitation est une exigence primordiale. Qui dit durée de vie de la batterie dit consommation d'énergie et délai de fonctionnement. On s'est donc posé la question de comprendre les processus qui assureront une consommation minimale d'énergie et un délai adéquat.

1.2 Objectif

Le but de ce travail est de faire une comparaison entre les protocoles ZigBee-Pro et ZigBee IP en termes de délai et de consommation d'énergie. Pour cela, on a utilisé différents outils matériels et logiciels pour pouvoir montrer tous les aspects de différence :

- Configuration et simulation des nœuds XBee-Pro (International-XBee, 2009) à l'aide de X-CTU (MaxStream, 2005).
- Implémentation d'un programme pour XBee dans le but de calculer le délai de transmission et de réception de l'information à partir d'un coordinateur dans le réseau.
- Manipulation à l'aide d'un outillage (en l'occurrence, d'oscilloscope) pour calculer l'énergie des nœuds capteurs.
- Utilisation du système d'exploitation Contiki OS (Sehgal, 2013) pour faire le test d'une application sur le simulateur de réseaux COOJA (Österlind et al 2010).

CHAPITRE II

ETAT DE L'ART

Dans ce chapitre, nous montrons le cadre théorique sur lequel nous nous sommes basés pour traiter les différents aspects de la norme ZigBee et les améliorations qui ont été faites depuis son lancement. Nous divisons ce chapitre en deux parties: la première est consacrée aux comparaisons des technologies sans fil déjà réalisées en ciblant plus le standard ZigBee, la deuxième partie montre les aspects de ressemblances et de différences entre ZigBee-Pro et ZigBee-IP déjà étudiées par les chercheurs qui se sont intéressés à ce sujet.

2.1 Comparaison de ZigBee et les autres technologies sans fil

Plusieurs recherches ont été faites sur les réseaux sans fil à courte portée et basse consommation d'énergie depuis leurs apparitions jusqu'à aujourd'hui. Pour mieux comprendre les spécificités de chacune des normes y compris ZigBee, (Baker, 2005) a fait une étude comparative pour mettre en valeur les points forts et faibles de ZigBee et Bluetooth dans les applications industrielles en attestant que la norme ZigBee basée sur IEEE802.15.4 est plus performante au niveau de la consommation d'énergie, la portée et la fiabilité de l'architecture maillée. Ils concluent que ZigBee est destiné à être une norme internationale ouverte, plus adaptée aux besoins industriels.

Dans l'étude de (Lee, Su, & Shen, 2007) sur les réseaux sans fil les plus utilisés (à savoir Bluetooth, UWB, ZigBee et Wi-Fi), on a évalué les principales caractéristiques

et comportements en termes de diverses mesures, y compris le délai de transmission, l'efficacité du codage des données, la complexité du fonctionnement et la consommation d'énergie. Ils ont abouti à des résultats qui affirment que ZigBee prend plus de temps par rapport aux autres normes puisqu'il a un faible débit maximal (qui est 250 Kbit/s). Le temps de transmission est proportionnel à la taille de la charge utile de données et disproportionné par rapport au débit. Or ZigBee consomme généralement moins d'énergie. On peut donc dire qu'il est mieux adapté aux applications mobiles des réseaux de capteurs sans fil fonctionnant sur batterie.

La comparaison qui a été faite dans cette étude ne valorise pas une des technologies discutées, mais aide surtout les ingénieurs dans la conception des nouvelles applications des réseaux sans fil.

L'analyse expérimentale du réseau ZigBee de (Pinedo-Frausto & Garcia-Macias, 2008) montre que ZigBee joue le rôle d'une simple commande entre le capteur et les nœuds ZigBee. Les couches hautes du réseau assurent deux types de routage en utilisant soit un maillage ou une arborescence; ce qui rend plus rapide le développement d'applications. En plus, la réutilisation des fonctionnalités, bibliothèques et profils existants facilite la construction de dispositifs dont ZigBee se sert fréquemment.

Les tests réalisés dans cette analyse prouvent que le débit est beaucoup plus faible (de l'ordre de 8.3kbit/s) que la valeur théorique, et que généralement la réception est considérablement plus lente que la transmission même si on fait varier la taille du message. Pour cela, l'Alliance ZigBee (Alliance, 2008) a mis l'accent sur le développement du standard pour répondre adéquatement aux besoins des applications de domotique et pour la réalisation de l'interopérabilité entre les dispositifs de différents fournisseurs.

Dans l'évaluation de (Kumar & Singh, 2011), les trois topologies possibles du standard ZigBee sont présentées et discutées au niveau de leur performance. Un réseau 802.15.4/ ZigBee IEEE nécessite au moins un dispositif fonctionnant en tant que coordinateur du réseau. Les autres dispositifs d'extrémité peuvent être réduits dans le but de diminuer le coût du système. La topologie maillée a plus d'avantage d'après cette évaluation. Les auteurs ont montré des exemples de routage maillé de la norme 802.15.4 basée sur les WSN. Le protocole AODV (*Ad hoc On Demand Distance Vector*) (Oumsis & al., 2010) est un protocole de routage destiné aux réseaux mobiles et DSR (Dynamic Source Routing) (Kanthé & al., 2012) un protocole de routage pour les réseaux maillés, donnent un délai de transmission optimal par rapport à d'autres protocoles de routage.

Pour mieux connaître les facteurs qui affectent les performances du réseau ZigBee, (Maurizio & al., 2010) ont fait une étude de propagation de fréquences radio des dispositifs ZigBee. Ils ont utilisé deux capteurs dont un coordinateur et un routeur du module ZB01CA produit par Digi. Le but était d'analyser leur comportement réel à l'intérieur d'un environnement hospitalier. Les mesures de la puissance reçue du signal radio (RSSI) montrent qu'il y a des facteurs internes comme les antennes intégrées et externes tels que le type de la propagation (par trajet multiple, évanouissement, etc.) et qui influencent sur le comportement du réseau.

Les expériences sont réalisées dans un laboratoire réel de travail non exempté de bruit. Ils concluent que l'émetteur et le récepteur doivent être en position de vue directe (Line of Sight) pour minimiser les pertes. Aussi la propagation par trajets multiples se considère stable dans ce genre d'environnement. Cette étude a été effectuée pour vérifier les exigences de performance d'un réseau de capteur sans fil mis dans un contexte réel dans le but d'estimer les principaux facteurs qui nuisent à son efficacité.

2.2 Travaux réalisés sur ZigBee-Pro et ZigBee-IP

Le travail effectué par (Franceschinis & al., 2013) compare les deux piles de protocole ZigBee-Pro et ZigBee IP développés dans le but d'intégrer ZigBee dans le mode d'Internet des Objets (*Internt of Things*).

ZigBee-Pro est un standard des réseaux de capteurs sans fil à faible puissance construit au-dessus de la norme IEEE802.15.4, héritant les mêmes couches PHY et MAC. Les spécifications des couches réseau et applications sont restées plus ou moins comme dans ZigBee original. La couche réseau gère la configuration, l'authentification, la découverte des voisins, le chemin optimal ainsi que le chiffrement en utilisant le standard AES à 128-bit. La couche application est divisée en sous-couche application (APS), application Framework (AF) et les ZigBee device objects (ZDO).

ZigBee-Pro ajoute des fonctionnalités au niveau du routage comme le routage *multi-cast* et *many-to-one* ainsi que les mécanismes de sécurité avancés comme l'échange de clés et les clés symétriques. La passerelle ZigBee joue un rôle primordial car elle expose les nœuds ZigBee vers l'internet.

L'objectif principal de ZigBee IP est d'utiliser la pile TCP/IP dans la norme ZigBee. La couche intermédiaire 6LoWPAN (Keng & al., 2010) permet aux couches physiques et MAC du standard 802.15.4 de s'intégrer facilement dans le monde IP via IPv6 permettant à tout objet doté d'interface de communication de faire partie du réseau. Conceptuellement comme 6LoWPAN, d'autres couches d'intégration ont été développées pour apporter des technologies sans fil de courte portée dans l'environnement IP, tel que le protocole CoAP (*Constrained Application Protocol*) (Bormann et al., 2012) qui est une alternative légère et plus simple que HTTP, basée sur UDP. CoAP utilise le style d'architecture REST (*Representational state transfer*) qui fournit un modèle d'interaction client/serveur entre les extrémités de

l'application, comprends des concepts clés sur le web ce qui rend facile l'intégration de http avec le web.

Le but de cet article est la comparaison des performances entre ZigBee-Pro et ZigBee IP. Ils se diffèrent dans la façon dont l'intégration est obtenue et où elle est positionnée.

Le protocole 6LoWPAN connaît un grand succès depuis qu'il a vu le jour car il assure l'intégration des réseaux de capteurs au monde IP. Sa conception et son architecture sont présentées dans (Mulligan, 2007).

L'approche de comparaison des différentes implémentations de 6LoWPAN sur WSN faite par (Mazzer et Tourancheau, 2009) a mis en valeur la technologie sans fil qui est devenue le premier moyen de communication pour les réseaux de capteurs. Afin de se conformer aux normes actuelles et futures et pour assurer l'accessibilité des nœuds capteurs, l'utilisation de l'IPv6 est inévitable. Étant donné qu'un nœud doit être économique énergiquement, ses capacités de calcul de puissance et du stockage doivent être limitées. C'est pour surmonter ces problèmes que le protocole 6LoWPAN a été conçu.

Plusieurs systèmes d'exploitation ouverts dédiés spécialement pour les réseaux de capteurs ont été comparés dans ce document, les auteurs ont choisi 'TinyOS' pour leur implémentation. Le tableau ci-dessous compare brièvement ces systèmes d'exploitation avec leurs options et caractéristiques. La dernière colonne indique s'il y a une mise en œuvre 6LoWPAN disponible.

TAB. 2.1: Exemples des Open sources des réseaux de capteurs sans fil (Mazzer et Tourancheau, 2009)

	TinyOS	Contiki	FreeRTOS	Mantis
Licence Schedul.	BSD run2comp FIFO ev preemp	BSD run2comp threads	mod GPL RT	BSD t-slot r-rob prio opt sleep m-thread
Footprint 6lowpan	0.4ko Yes	20ko Yes	256ko Yes	0.5ko No
	BTnut	SOS	NanoRK	Dream
License Schedul.	BSD m-thread	mod BSD kern mod hot-plug	Qt-like RT	n.c. ev-driv m-thread
Footprint 6lowpan	n.c. No	1.5ko No	2ko No	3.5ko No

Les piles 6LoWPAN existantes se coïncident comme des logiciels monolithiques, cela ne permet pas leur modification. Afin d'exploiter toutes les capacités de TinyOS et d'être en mesure d'évoluer dans les choix de mise en réseau, les auteurs ont développé leur propre pile.

La domotique constitue un vrai profit pour les technologies basées sur IPv6 et 6LoWPAN. Elles permettent d'utiliser les infrastructures existantes des réseaux locaux réduisant ainsi les coûts d'installation. 6LoWPAN étend ce réseau et supporte la technologie Web; ce qui facilite le développement et garantit la facilité d'exploitation des ressources existantes. Une comparaison de certaines technologies concurrentes pour l'automatisation de la maison est fournie dans (Kovatsch et al., 2010) et résumée dans le TAB. 2.2.

TAB. 2.2: Comparaison des solutions pour les maisons automatisées (Kovatsch et al., 2010)

	X10	KNX	ZigBee	dS	IPv6
Medium	PLC, RF 310MHz (US), 433MHz (EU)	TP, RF 868MHz	RF 2.4GHz, 868MHz (EU), 915MHz (US)	PLC	Ethernet, Wi-Fi, RF 2.4GHz
Network size	2^8	2^{16}	2^{16}	2^{16}	2^{64} per subnet
Data rate	20b/s	9.6kb/s	20..250kb/s	200b/s	250kb/s..1Gb/s
Interface	custom solutions	application level gateway	application level gateway	Web services	UDP, TCP, RESTful Web
Maturity	1975	2002 (1990)	2004	2010	1998 (1969)
Costs	low	high	medium	medium	low
Installation overhead	low	high	low	medium	low
Connectivity	low	medium	medium	medium	high
Security	none	high (EIBsec)	medium (AES)	low (private circuits)	medium (6LoWPAN AES only)

Un autre exemple de maison intelligente est abordé dans l'article de (Zou et al., 2011) décrit la façon dont ZigBee s'intègre avec l'IPv6. La première étape été de mettre la pile IPv6 au-dessus de la couche réseau du standard ZigBee. Tous les nœuds ZigBee sont assignés à une adresse IPv6. La deuxième approche est la conception de couche à double pile 6LoWPAN et ZigBee fonctionnant sur la base de la couche MAC de la norme 802.15.4. La troisième étape est l'ajout d'une passerelle qui permet à ZigBee et 6LoWPAN de s'adapter à IPv6 parallèlement. Ce genre d'application adopte souvent une topologie en étoiles, qui est composé de passerelles et plusieurs nœuds de capteurs ZigBee. La passerelle agit comme un coordinateur du réseau pour mettre en place le réseau et gérer son fonctionnement à distance. Les terminaux mobiles les téléphones mobiles sont gérés par IPv6 qui aide à la mobilité.

Le document présenté par (Toscano et Lo, 2012) aborde l'adaptation des mécanismes de la norme 802.15.4/ZigBee et du protocole 6LoWPAN pour fonctionner à faible puissance. Généralement, ces mécanismes mettent des nœuds en veille de façon à diminuer le cycle de service. Bien que les deux technologies sont basées sur les mêmes couches inférieures mais l'approche de ZigBee est tout à fait différente de celle adoptée par 6LoWPAN. D'une part ZigBee définit le mode de balise (*beacon*)

qui permet aux nœuds de rester en veille (mode faible puissance) pour la plupart du temps d'où le coordinateur envoie périodiquement aux nœuds une balise de synchronisation tandis que la transmission avec nœuds dormants est obtenue par l'utilisation du routage en arbre. D'autre part, le protocole 6LoWPAN basé sur IPv6 est asynchrone. Il adopte une topologie de maillage et utilise un algorithme de routage qui ne prend pas en charge les nœuds dormants; ce qui nécessite des mécanismes d'écoute de canal pour des fins d'économie d'énergie.

La comparaison des performances des deux protocoles montre que ZigBee offre des cycles de services plus courts et fournit le plus petit délai maximal de bout-en-bout, or 6LoWPAN assure une fiabilité plus élevée. Il fournit le plus petit délai de bout-en-bout avec en plus un faible pourcentage de perte de paquets. Les deux protocoles prennent généralement la même durée pour se mettre à jour.

Les diverses mises en œuvre de 6LoWPAN actuellement disponibles sont soit dépendantes des systèmes d'exploitation utilisés sur les nœuds capteurs, soit embarqués dans des produits commerciaux (Pediredla et al., 2013). Des exemples existants d'implémentations de 6LoWPAN sont donnés dans le TAB. 2.3.

TAB. 2.3: Implémentations 6LoWPAN existantes (Pediredla et al., 2013)

6LoWPAN stack	Free or Commercial
BLIP on TinyOS	Free
NanoStack v2.0	Commercial
SICSlowpan on Contiki	Free
NanoStack v1.1.0 (on FreeRTOS)	Free

La convergence de l'IPv6 et des réseaux de capteurs sans fil à faible puissance est pragmatique et très efficace surtout pour le domaine des systèmes embarqués d'après la vision de (Hui et Culler, 2010). Cette convergence semble être économique, mis à part quelques défis et limites d'architecture IP. Les progrès qui ont été faits dans ce domaine lui permettront de rejoindre les rangs des technologies à succès comme Ethernet et Wifi.

CHAPITRE III

TECHNOLOGIES UTILISÉES

3.1 Le standard ZigBee

Le réseau ZigBee assure la transmission des données sans fil de haut niveau dans le but de faire communiquer des petits périphériques à travers des liaisons radio. Il définit un ensemble de protocoles de communication pour un débit théorique de 250kbit/s et garantit une communication de courte distance dans les environs de 100m tout en étant moins cher et plus simple.

ZigBee opère dans la bande de fréquence 2.4 GHz mettant en œuvre la modulation OQPSK (*Offset quadrature phase-shift keying*) et la technique d'étalement spectral DSSS (*Direct Sequence Spread Spectrum*).

La modulation OQPSK est une forme de déplacement de phase par décalage dans laquelle quatre angles de phase différents sont utilisés. C'est une variante de la modulation QPSK qui transmet 4 valeurs de phase (2 bits à la fois). Les quatre angles sont séparés par un saut de 180° . Ces déphasages se traduisent par les fluctuations importantes d'amplitude. En décalant la synchronisation des bits pairs et impairs d'une période ou d'une demi-période de symbole, les composantes en phase et en quadrature ne changeront pas en même temps. La modulation OQPSK est donc intéressante pour les systèmes de communication mobile car l'efficacité de la bande passante et les amplificateurs non linéaires sont essentielles pour une faible consommation d'énergie.

La technique d'étalement de spectre à séquence directe DSSS fait la transmission de signal dans lesquelles l'énergie émise avec une ou plusieurs harmoniques est délibérément étalée ou distribuée dans le domaine fréquentiel. Elle diminue le risque d'interférences avec d'autres signaux parasites tout en garantissant une certaine confidentialité.

ZigBee possède les spécifications mentionnées dans le TAB. 3.1.

TAB. 3.1: Spécifications du standard ZigBee

Parameter	Specification
Data rate	250 kbps
Number of channels	16
Operating frequency	2.4GHz
Channel spacing	5 MHz
Spread spectrum	Direct Sequence Spread Spectrum
Chip rate	2 Mega chips per second.
Modulation	OQPSK with Half sine pulse shaping

3.1.1 WSN et le standard ZigBee

Un WSN permet à des dispositifs de communiquer pour surveiller des événements. Ces dispositifs sont généralement des capteurs transforment une grandeur physique en une autre mesurable afin d'en faire une estimation après l'avoir acheminé à l'utilisateur concerné.

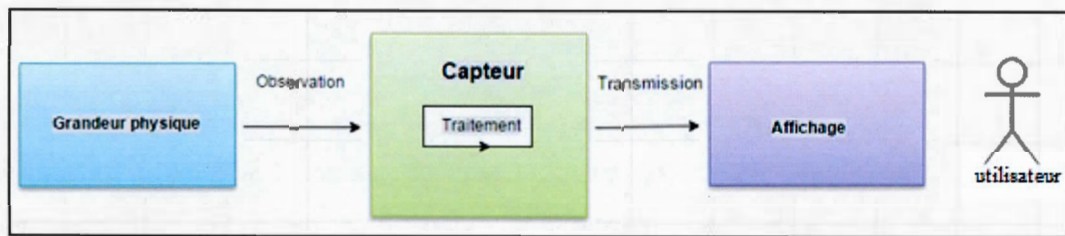


FIG. 3.1: Schématisation du rôle d'un capteur

Ces capteurs mesurent les changements, puis envoient les données collectées à un coordinateur qui supervise l'ensemble de capteurs, échange les données avec d'autres réseaux et fait le lien avec l'internet.

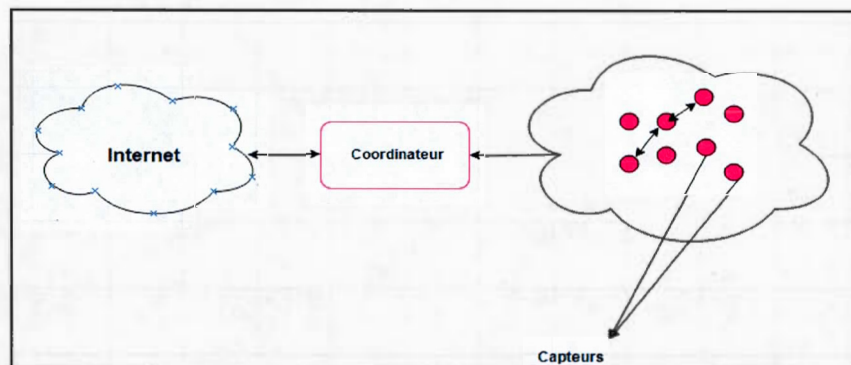


FIG. 3.2: Architecture d'un WSN

3.1.2 IEEE 802.15.4/ ZigBee

Le 802.15.4 est un protocole de communication défini par IEEE et est dédié aux réseaux personnels sans fil à bas débit (LR-WPAN Low Rate-Wireless Personal Area Network). L'alliance ZigBee a été créée dans le but de promouvoir ce protocole. Aussi pour développer des nouvelles applications ZigBee facile à utiliser qui répondent aux besoins du marché dans divers domaines tels que la domotique, l'environnement, la santé, etc. Le standard ZigBee s'appuie sur l'IEEE 802.15.4 pour la couche physique et la couche MAC.

3.1.3 Architecture du standard ZigBee

L'architecture du standard ZigBee est composée de 4 couches: physique, liaison ou (MAC), réseau et application :

- **Couche physique :** traduit les trames envoyées et reçues en bits et gère l'utilisation de la transmission radio et la communication des canaux. La norme prévoit deux couches physiques différentes, une pour le 868/915MHz et une seconde pour le 2,4 GHz en utilisant la modulation OQPSK à spectre étalé à séquence discrète (DSSS).

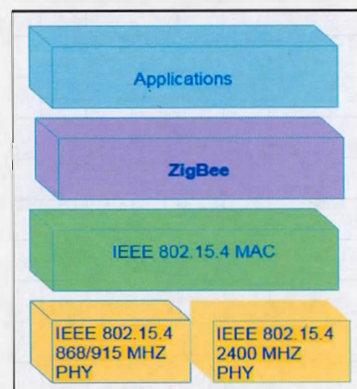


FIG. 3.3: Architecture du standard ZigBee

- **Couche MAC :** gère la transmission des trames et des slots de temps réservés, contrôle l'accès au médium sans fil en utilisant le mécanisme CSMA/CA (Francomme,2008), synchronise le réseau et assure la fiabilité des communications en utilisant des CRC et des retransmissions.
- **Couche réseau (ZigBee) :** définie par l'Alliance ZigBee. Cette couche assure le routage, envoie/reçoit les données vers et depuis la couche application, construit et maintient la topologie du réseau, fait l'adressage des nœuds (en

supportant jusqu'à 65536 nœuds), fournit les mécanismes pour joindre, quitter et former un réseau et finalement garantit la sécurité.

- Couche application : la couche la plus haute de la pile protocolaire. Elle détermine la façon dont toutes les couches inférieures se comportent. Elle est associée à plusieurs modules comme : SSP (Security Service Provider) qui gère les fonctions de sécurité (authentification, cryptage); APS (Application Support Sub-Layer) qui soutient la mise en liaison des dispositifs et les services de messagerie et ZDO (ZigBee Device Object) qui permet la découverte des dispositifs et des services (Val et al.,2008).

3.1.4 Dispositifs du standard ZigBee

Il existe deux types de dispositifs du standard :

- FFD (full fonction device): actif et toujours à l'écoute, il est alimenté par une source d'énergie inépuisable, a des capacités avancées comme la transmission des paquets et assure la synchronisation des nœuds.
- RFD (Reduced fonction device): communique seulement avec les dispositifs FFD. Ses fonctionnalités sont limitées. Il fournit peu de tâches telles que l'acquisition des données (Ex. données de température et le contrôle des dispositifs externes). Il est presque toujours à l'état dormant.
- Coordinateur ZigBee (ZC) : considéré comme un FFD. C'est l'élément qui forme la racine et maintient le reste du réseau. Chaque topologie doit avoir un seul coordinateur qui stocke les clés de sécurité et gère les données.

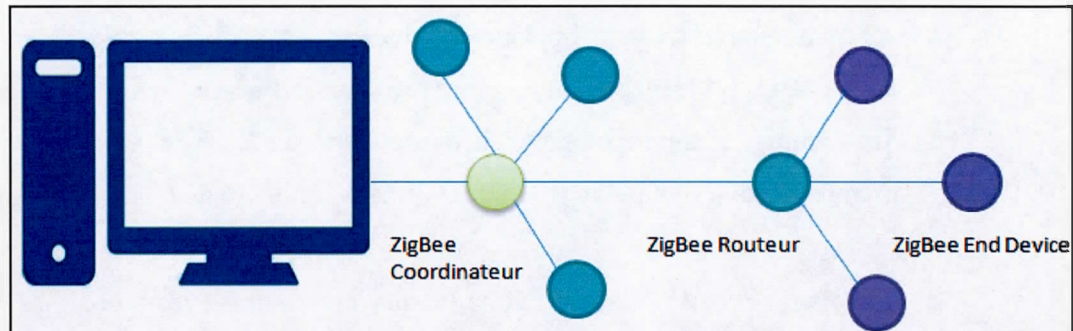


FIG. 3.4: Dispositifs du standard ZigBee

- **Routeur ZigBee (ZR) :** considéré aussi comme FFD, il joue le rôle d'intermédiaire qui achemine ou transmet les données entre les appareils. Il aide à étendre la portée physique d'un réseau Zigbee en permettant à plusieurs nœuds de se joindre au réseau.
- **End Device :** peut être un FFD ou RFD dépendamment de l'application. C'est l'interface physique de chaque capteur.

3.1.5 Topologies du réseau

Un réseau ZigBee peut être organisé selon une des trois topologies suivantes :

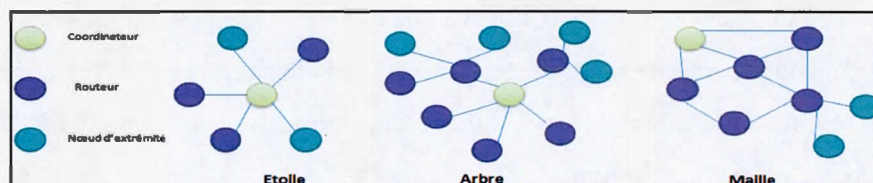


FIG. 3.5: Topologies du réseau ZigBee

- **Topologie étoile :** Cette topologie se compose principalement du coordinateur centralisé. Les routeurs et les nœuds d'extrémités ne peuvent communiquer qu'avec lui. Cette topologie peut ne pas être adéquate pour les réseaux de

capteurs sans fil en raison de l'énorme consommation d'énergie par le coordinateur ZigBee.

- **Topologie maille :** C'est une topologie décentralisée. Les appareils peuvent communiquer entre eux. Lorsque certaines routes sont en panne, l'algorithme de routage est capable de prendre un autre itinéraire parmi les options disponibles. Cette topologie consomme moins d'énergie par rapport à la première. Son principal inconvénient est la complexité de la mise en œuvre du routage.
- **Topologie arbre :** c'est un cas particulier qui combine les topologies étoiles et maille. Un dispositif d'extrémité se joint à l'arbre par l'intermédiaire d'un routeur. Ce dernier se joint à l'arbre par l'intermédiaire d'un autre routeur (le coordinateur ZigBee peut être utilisé aussi en tant que routeur). Chaque routeur maintient la synchronisation des échanges de données des dispositifs. Le reste du temps, les ends device peuvent économiser de l'énergie en mode veille.

3.2 ZigBee-Pro

La version ZigBee-Pro est une amélioration significative du standard ZigBee d'origine. Certaines caractéristiques optionnelles de ZigBee sont devenues obligatoires dans ZigBee-Pro qui maximise les capacités radio de la norme IEEE 802.15.4. C'est un support avancé pour les grands réseaux constitués de d'un grand nombre de nœuds. Il permet la transmission de messages volumineux en utilisant la fragmentation et le réassemblage. Il assure la sécurité, l'adaptation et l'autoréparation du réseau. Une passerelle est utilisée comme moyen de transfert des données des nœuds ZigBee vers Internet.

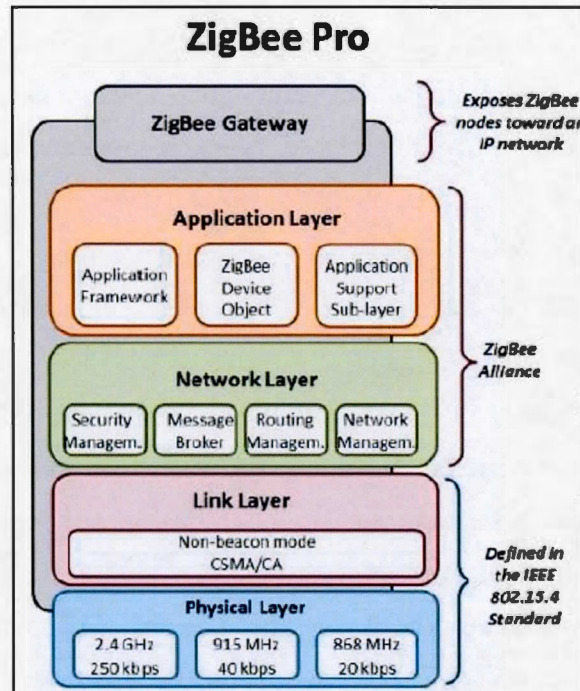


FIG. 3.6: Architecture du ZigBee-Pro (Franceschinis et al.,2013)

ZigBee-Pro est basé sur une topologie maillée et un réseau sans balise (non beacon mode). Ses caractéristiques sont :

- **Adressage** : les nouveaux dispositifs qui s'ajoutent au réseau auront automatiquement une adresse au hasard reconnue une fois que l'appareil est synchronisé. Si deux nœuds ont la même adresse, alors la couche réseau intervient on se basant sur l'adresse MAC de chaque nœud.
- **Gestion des liens** : chaque nœud de la topologie maillée est capable de communiquer avec son voisin en évaluant sa qualité et en choisissant le meilleur pour transmettre le paquet.
- **Changement de canal** : lorsque plusieurs nœuds ZigBee détectent de l'interférence due au chevauchement ou au bruit, le gestionnaire du canal sélectionne un nouveau canal et avise le reste du réseau.

- Fragmentation : ZigBee-Pro fournit la fragmentation des grands paquets en plus petits de telle façon qu'ils soient faciles pour l'envoi. Le nœud de destination réassemble ensuite les paquets.
- Gestion de l'alimentation: ZigBee-Pro permet aux périphériques d'extrémité de se mettre en mode veille afin qu'ils consomment moins d'énergie. Lorsqu'ils se réveillent, ils se mettent à jour dans le cas de changement dans le réseau.
- Routage: il y a deux types de routage offerts par ZigBee-Pro: routage multicast et many-to-one. Ce dernier est utilisé pour un réseau avec concentrateur.
- Il existe deux modes de sécurité dans ZigBee-Pro :
 - Mode standard : les dispositifs sont autorisés à utiliser la clé du lien et la clé du réseau qui standard pour tous, ils ne sont pas obligés de s'identifier pour se joindre à un réseau.
 - Mode de sécurité supérieure : Les dispositifs sont obligés d'effectuer une authentification avec leur 'parent' et entre 'voisins'. Trois types de clés existent: clés du réseau, du lien et du coordinateur (F. Shanin, 2008).

3.3 ZigBee IP

ZigBee IP est considéré comme le premier standard ouvert destiné aux réseaux maillés sans fil à faible débit et courte portée, compatible avec le protocole IPv6. Il assure l'établissement des connexions Internet de manière transparente avec des dispositifs à faible puissance et faible coût.

3.3.1 Architecture de ZigBee IP

La pile protocolaire ZigBee IP basée également sur les couches physiques et MAC de la norme IEEE802.15.4, améliore les couches réseau et application conformément aux recommandations de l'IETF (Internet Engineering Task Force). L'utilisation des protocoles standards tels que IPv6, 6LoWPAN, RPL, TCP/UDP consiste à avoir une

architecture extensible censée d'être intégrée dans un cadre de l'Internet des objets puisqu'elle assure un réseautage IPv6 de bout en bout sans faire appel aux passerelles intermédiaires. Elle offre une sécurité grâce au protocole TLS (Transport Layer Security) qui fournit l'authentification du serveur, la confidentialité et l'intégrité des données échangées. Elle offre aussi une sécurité au niveau de la couche Liaison de données via l'algorithme AES-128 et la suite de chiffrement ECC-256.

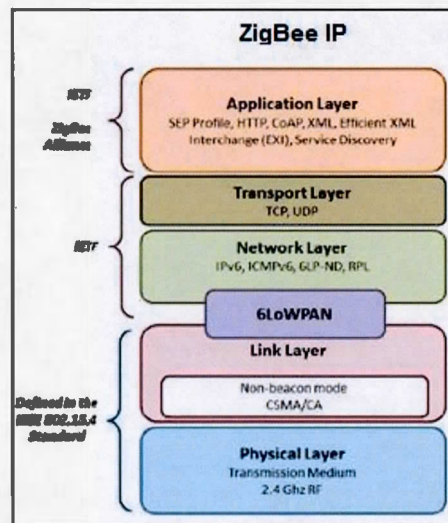


FIG. 3.7: Architecture du réseau ZigBee IP (Franceschinis et al., 2013)

3.3.2 La technologie 6LoWPAN

La couche d'adaptation 6LoWPAN garantit l'interopérabilité entre la couche liaison définie par la norme 802.15.4 et la couche réseau qui permet l'utilisation efficace d'IPv6 dans le but de lier n'importe quel appareil à l'internet. Cette norme a mené à la normalisation de 6LoWPAN vu ses caractéristiques. Bien qu'il soit de courte portée, de faible puissance et de débit limité, il s'adapte bien aux dispositifs qui sont petits et simples. La communication IPv6 est volumineuse par rapport à des dispositifs de ce type comme les nœuds capteurs. Ce qui nécessite des mécanismes de compression et de fragmentation de données pour rendre cette adaptation possible.

3.3.2.1 Architecture 6LoWPAN

L'envoi des paquets dans une topologie 6LoWPAN commence par le dispositif d'extrémité RFD qui achemine le même paquet à un nœud routeur FFD qui transmet, saut par saut le paquet à la passerelle 6LoWPAN. Cette passerelle qui est connectée à internet avec le domaine IPv6 transmettra ensuite le paquet à destination en utilisant le protocole IP.

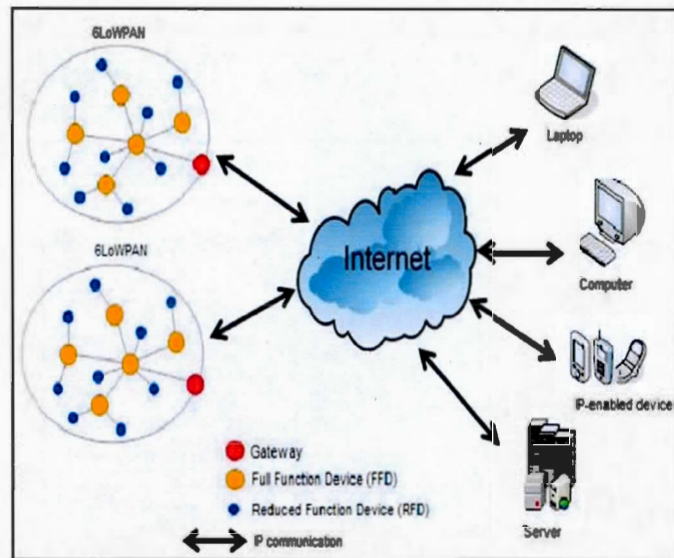


FIG. 3.8: Architecture du protocole 6LoWPAN(Keng et al., 2010)

3.3.2.2 Mécanismes de fragmentation et réassemblage 6LoWPAN :

L'unité de données (i.e. la trame) de la couche physique de la norme IEEE 802.15.4 est de 127 octets. En tenant compte de 25 octets de la sous-couche MAC, il reste 102 octets de charge utile pour la couche Liaison de données. Pour des fins de sécurité imposée par cette couche, la taille du paquet est encore réduite à 81 octets pour le niveau IP dans les meilleurs des cas. De plus, les tailles des en-têtes TCP et UDP sont de 20 et 8 octets respectivement sans oublier la taille de l'en-tête IPv6 qui est de 40 octets. Finalement, l'espace des données disponible au niveau de la couche

application est de 21 octets pour TCP et 33 octets pour UDP. Ce qui ne respecte pas les spécifications de l'IPv6 (un MTU minimal doit être de 1280 octets).

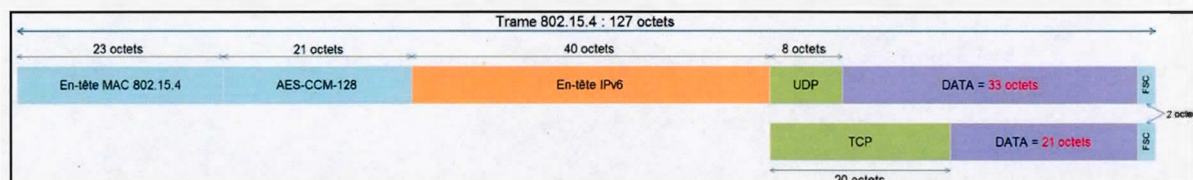


FIG. 3.9: taille de la trame de la norme IEEE 802.15.4

Un codage HC1 a été défini par 6LoWPAN. C'est un système de compression optimisé pour la communication IPv6. Les champs de longueur et d'adresse IPv6 sont éliminés tant que la couche d'adaptation peut les obtenir à partir de l'en-tête de la couche liaison de données. Les autres champs sont compressés à quelques bits pour réduire les transmissions. Le TAB. 3.2 compare les deux en-têtes IPv6 et 6LoWPAN :

TAB. 3.2: Comparaison des en-têtes IPv6 et 6LoWPAN (Keng et al., 2010)

Header Field	IPv6 header length	6LoWPAN HC1 length	Explanation
Version	4 bits	-----	Assuming communicating with IPv6.
Traffic class	8 bits	1 bit	0 = Not compressed. The field is in full size. 1 = Compressed. The traffic class and flow label are both zero.
Flow label	20 bits		
Payload length	16 bits	-----	Can be derived from MAC frame length or adaptation layer datagram size (6LoWPAN fragmentation header).
Next header	8 bits	2 bits	Compressed whenever the packet uses UDP, TCP or Internet Control Message Protocol version 6 (ICMPv6).
Hop limit	8 bits	8 bits	The only field always not compressed.
Source address	128 bits	2 bits	If Both source and destination IPv6 addresses are in link local, their 64-bit network prefix are compressed into a single bit each with a value of one. Another single bit is set to one to indicate that 64-bit interface identifier are elided if the destination can derive them from the corresponding link-layer address in the link-layer frame or mesh addressing header when routing in a mesh.
Destination address	128 bits	2 bits	
HC2 encoding	-----	1 bit	Another compression scheme follows a HC1 header.
Total	40 bytes	2 bytes	Fully compressed, the HC1 encoding reduces the IPv6 header to two bytes.

La couche d'adaptation reçoit le paquet IPv6 de 1280 octets. Elle doit ensuite l'envoyer dans des trames 802.15.4 qui ne contiennent que 81 octets d'espace libre. Ce paquet doit donc être fragmenté avant l'envoi et réassemblé à la réception. La première fragmentation est de 5 octets et comprend 1 octet pour le champ de décalage avec la valeur '11000', 11 bits pour la taille du paquet avant la fragmentation et 16 bits comme identifiant commun à tous les fragments d'un même paquet. À partir du deuxième fragment, 8 bits de 'Datagram Offset' sont ajoutés pour indiquer la position du fragment dans le paquet IP.

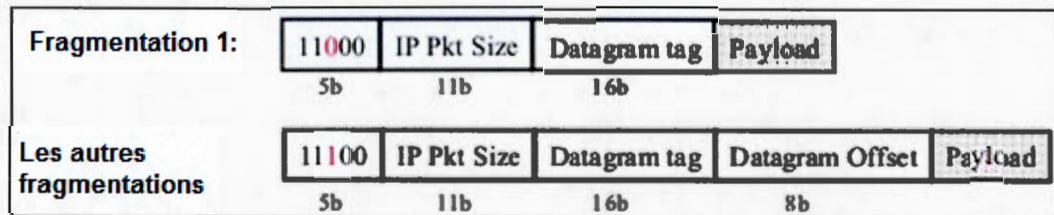


FIG. 3.10: méthode de fragmentation 6LoWPAN

3.3.2.3 : Les couches OSI/6LoWPAN

La comparaison des couches du modèle OSI et les couches du protocole 6LoWPAN nous permettent de montrer le positionnement de la couche d'adaptation et la nécessité de fragmenter et de réassembler les paquets de données entre deux capteurs sans fil.

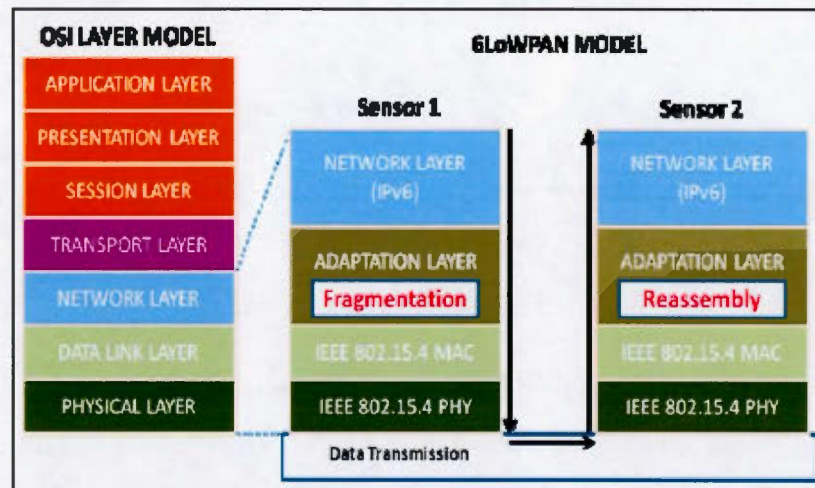


FIG. 3.11: Comparaison entre deux modèles de couches OSI et 6LoWPAN (Ghazali et al., 2012)

3.3.3 Le protocole CoAP

Le protocole d'application pour l'internet des objets (CoAP) a été créé récemment par le groupe (Constrained RESTful Environment Core) de l'IETF. Ce protocole logiciel construit sur UDP est destiné à être utilisé dans des dispositifs de capacités limitées et peu fiables comme les nœuds des réseaux de capteurs sans fil.

Le CoAP est basé sur un style d'architecture REST (Representational State Transfert) utilisant des spécifications originelles du protocole HTTP telles que l'URI et les réponses HTTP comme représentation des ressources sous différents formats (HTML, XML, JSON, etc.) L'architecture CoAP se compose de deux couches :

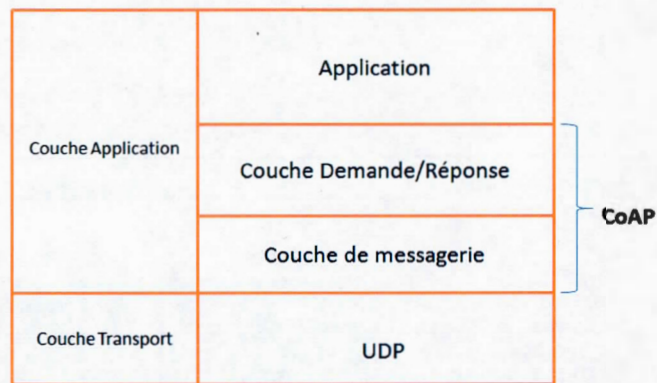


FIG. 3.12: Architecture du protocole CoAP

- Couche de messagerie: gère la non-fiabilité des messages sur UDP entre deux points d'extrémité ainsi que la nature asynchrone des interactions. Il existe quatre types de messages: confirmable (CON), non-confirmable (NON), acquittement (ACK), et remis à zéro (RST).
- La deuxième couche: fournit un modèle d'interaction demande/réponse et comprend mêmes requêtes HTTP GET, POST, PUT et DELETE.

3.4 Outils de simulation : Contiki OS/ Cooja

Contiki est un système d'exploitation embarqué, flexible et léger, destiné aux réseaux de capteurs. Il fournit une pile complète basée sur IPv4 et IPv6 ainsi que d'autres fonctionnalités intéressantes telles que RPL, 6LoWPAN, CoAP, etc.

Développé récemment par une équipe de recherche suédoise, Contiki connaît un grand succès et intéresse de plus en plus de développeurs. Disponible gratuitement sous licence BSD, il peut être utilisé ou modifié, même pour des fins commerciales.

Composants de communications dans Contiki

- La consommation d'énergie étant un facteur clé dans les réseaux de capteurs, Contiki intègre une couche RDC (Radio Duty cycling) sous la couche MAC.

Ceci a pour but de réduire la puissance, minimiser les collisions et optimiser la consommation d'énergie.

- La couche Rime supporte les opérations simples telles que l'envoi d'un message à un capteur voisin. Les utilisateurs peuvent choisir d'utiliser cette couche légère de Contiki dans le cas où une mise en œuvre de capteurs n'a pas besoin de la pile IPv6. Cela réduira l'utilisation des ressources.
- La couche UIPv6 orientée internet offre les services IP ainsi que les protocoles TCP et UDP intégrés dans le système d'exploitation Contiki. Elle permet aussi d'interagir directement avec les serveurs web, mais nécessite plus de ressources de la couche Rime.

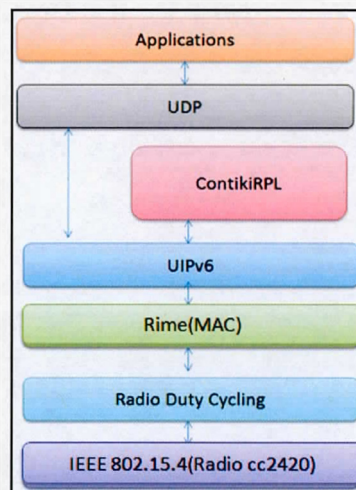


FIG. 3.13: Pile protocolaire Contiki

Le simulateur de réseau Cooja :

Contiki fournit également un simulateur de réseau afin d'accélérer le développement et les tests des applications. Le simulateur Cooja permet de visualiser l'utilisation de l'énergie et de la radio des réseaux de capteurs ainsi que plusieurs autres paramètres.

CHAPITRE IV

RÉALISATION ET SIMULATION

Dans ce chapitre, nous montrons les réalisations matérielles et logicielles faites dans le cadre de la comparaison entre les deux versions ZigBee-Pro et ZigBee IP. Une première partie est dédiée aux nœuds XBee. Elle nous permettra de calculer le délai de transmission des données ainsi que la consommation d'énergie des nœuds capteurs. La deuxième partie est sous forme d'une application basée sur le système d'exploitation Contiki qui supporte les standards 6LoWPAN et CoAP avec lequel nous simulons des topologies du réseau ZigBee IP.

4.1 Réalisation matérielle du réseau ZigBee-Pro

4.1.1 Le module XBee :

XBee est un transceiver (émetteur/récepteur) de petite taille qui s'alimente en 3.3v, implémentant le protocole ZigBee. Il est bidirectionnel, peut aussi bien émettre et recevoir des données alternativement (half-duplex).

Il existe en deux versions XBee et XBee-Pro. La principale différence entre les deux réside dans la puissance d'émission.

Ce module à faible coût se connecte facilement au moyen d'une liaison UART (Universal Asynchronous Receiver Transmitter) entre l'ordinateur et le port sériel classique ou sur le port RS-232 (port COM) à une vitesse atteignant 115200 bauds. Les entrées/sorties de XBee sont montrées sur la FIG. 4.1.

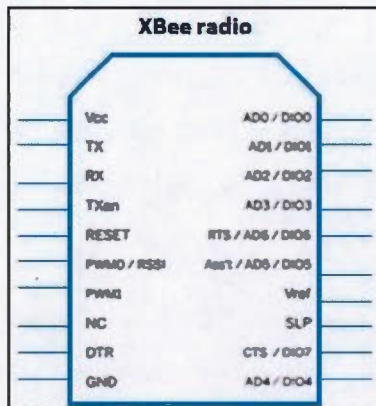


FIG. 4.1: Entrée/sortie du module XBee

Les entrées/sorties DIO sont numériques et les entrées AD sont analogiques (échantillonnage des tensions analogiques converties en numérique). Les sorties PWM et PWM1 sont analogiques.

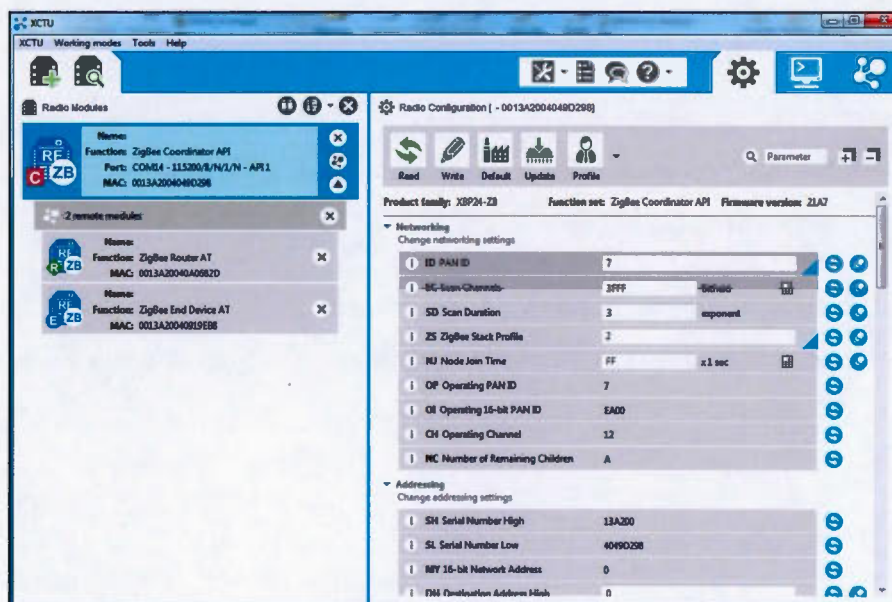


FIG. 4.2: Configuration des nœuds XBee

4.1.2 Configuration des nœuds du réseau

Le logiciel X-CTU est utilisé pour configurer les XBee avant leurs intégrations dans le réseau. Il est aussi utilisé pour choisir ou modifier leur mode de fonctionnement.

Dans notre travail, nous avons utilisé la nouvelle version de X-CTU (FIG. 4.2), qui dispose d'une interface plus sophistiquée. De plus, elle permet de configurer des XBee non connectés à l'ordinateur, mais via le réseau ZigBee.

La configuration du coordinateur permet grâce à une API l'envoi et la réception des paquets de données. Ce mode facilite la modification des paramètres en cas d'adaptation du réseau. Il est également utile si l'utilisateur a besoin de modifier la configuration d'un module distant. Les deux autres dispositifs sont configurés en mode AT. C'est un mode transparent qui exécute des commandes prédéfinies par le port UART.

Dans ce qui suit, nous allons présenter trois différentes topologies dans le but de comparer le délai de transmission de bout en bout et la consommation d'énergie.

4.1.3 Première topologie :

La première topologie réalisée sur la FIG. 4.3 consiste à faire communiquer les trois nœuds XBee, un coordinateur, un routeur et un nœud d'extrémité communiquant entre eux en utilisant le réseau ZigBee.

Le principe est de récolter les données envoyées par le capteur en passant par le PC1 qui joue le rôle de passerelle afin de visualiser ces mêmes données sur le PC2.

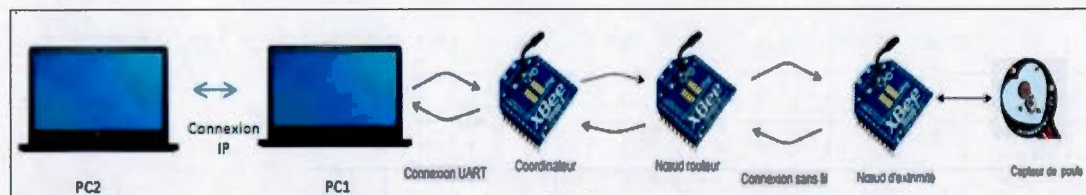


FIG. 4.3: Architecture du premier réseau ZigBee

Après avoir configuré les nœuds XBee du réseau ZigBee avec l'outil X-CTU, on a été en mesure de visualiser la connexion entre ces nœuds tels que montrés sur la FIG. 4.4 :

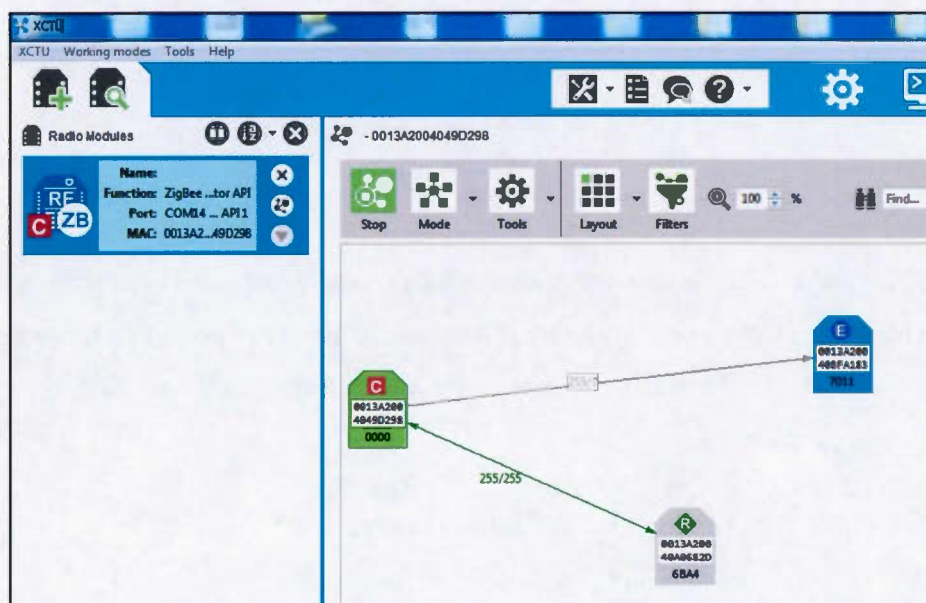


FIG. 4.4: Simulation du premier réseau avec X-CTU

Un capteur de pulsations du rythme cardiaque (poul) a été utilisé dans cette expérience et a été testé sur un être humain. Il envoie les données captées au nœud d'extrémité qui les achemine au coordinateur. Ce dernier se charge d'activer la connexion sans fil avec les autres modules du réseau, gère les données reçues et les

envoi sur un port du PC1 qui joue le rôle d'une passerelle (en l'occurrence le port 14). Ces données seront stockées sur le PC1 dans un fichier sous format JSON et téléchargées depuis son URL à partir du PC2. Ce processus prend un certain délai qui sera calculé (voir FIG. 4.5) grâce aux programmes implantés à cet effet.

```

value:1023 TIME:78 ms
value:1023 TIME:79 ms
value:1023 TIME:89 ms
value:1023 TIME:60 ms
value:1023 TIME:79 ms
value:1023 TIME:79 ms
value:1023 TIME:80 ms
value:1023 TIME:59 ms
value:1023 TIME:79 ms

Average:76

END.....

[main] INFO com.digi.xbee.api.XBeeDevice -
BUILD SUCCESSFUL (total time: 8 seconds)

```

FIG. 4.5: Délai de transmission des données (première topologie, expérience 1).

L'implémentation que nous avons effectuée est basée sur la librairie Java de XBee (FIG. 4.6). Cette application optimise considérablement l'utilisation des modules sans fil XBee et fournit différentes fonctionnalités dans le but de faciliter le développement :

- Le support de protocole ZigBee et modules XBee.
- La découverte des nœuds configurés.
- L'envoi et réception des données entre les nœuds XBee du réseau.
- La gestion des entrées/sorties des modules.

```

public class MainApp {

    private static final String PORT = "COM14";
    private static final int BAUD_RATE = 115200;

    static private RemoteXBeeDevice myED0;
    //static private RemoteXBeeDevice myED1;
    //static private RemoteXBeeDevice myED2;

    public static void main(String[] args) {
        System.out.println(" +-----+");
        System.out.println(" | XBee Java Library ZigBee application |");
        System.out.println(" +-----+");

        XBeeDevice coordateur = new XBeeDevice(PORT, BAUD_RATE);

        try {
            coordateur.open();
            coordateur.readDeviceInfo();
            System.out.println("getNodeID: " + coordateur.getNodeID() + "\n");

            System.out.println(" *** + coordateur.get64BitAddress().generateDeviceID() + *****\n");
            System.out.println(" *****\n");
            // Obtain the remote XBee device from the XBee network.
            XBeeNetwork xbeeNetwork = coordateur.getNetwork();
            xbeeNetwork.setDiscoveryTimeout(19000);
            xbeeNetwork.setDiscoveryOptions(EnumSet.of(DiscoveryOptions.APPEND_DD, DiscoveryOptions.DISCOVER_MYSELF));

            xbeeNetwork.startDiscoveryProcess();
            while (xbeeNetwork.isDiscoveryRunning()) {

```

FIG. 4.6: Un extrait du programme Java pour l'application XBee

Afin de montrer l'aspect IP dans les expériences réalisées pour ZigBee-Pro, Une autre implémentation a été réalisée pour relier le PC1 au PC2 via Ethernet. Les données récupérées du PC1 et stockées dans un fichier de format JSON seront téléchargées depuis leur URL avec un programme java à partir du PC2 en indiquant le chemin du fichier commençant par l'adresse IP de la passerelle «127.0.0.1 ».

La taille des fichiers varie dépendamment des données captées. Les données de la topologie 1 nommées '1ed' sont de 373 octets et sont téléchargés sur le PC2 pendant 0.03ms.


```

1 package com.test;
2
3 import java.io.File;
4
5
6
7
8
9 public class Download {
10
11     private static String OUTPUT_PATH = "C:/DEV/TEST/OUT";
12
13     private static String INPUT_PATH = "file:///127.0.0.1/test/led.json";
14
15     public static void getFile() {
16         InputStream input = null;
17         FileOutputStream writeFile = null;
18
19         try {
20
21             URL url = new URL(INPUT_PATH);
22             URLConnection connection = url.openConnection();
23             int fileLength = connection.getContentLength();
24             if (fileLength == -1) {
25                 System.err.println("Invalid URL or file.");
26                 return;
27             }
28         }

```

FIG. 4.7: Extrait du programme Java pour le téléchargement des données PC2

Nos expériences sont réalisées dans des conditions stables et constantes au sein du laboratoire TRIME de l'UQAM. Plusieurs tests ont été faits dans le but d'avoir des résultats exacts. Le TAB. 4.1 détermine le délai moyen de trois expériences de la topologie 1 réalisée:

TAB. 4.1: Délai moyen pour la topologie1

	Experience1	Experience2	Experience3	Moyenne (ms)
Topologie 1	76,03	81,03	80,03	79,03

L'énergie consommée par un nœud capteur est généralement dû à l'observation, le traitement et la transmission de données. L'avantage des nœuds capteurs XBee est qu'ils peuvent se mettre en état de veille dans le cas où aucun événement ne survient.

Pour le calcul de l'énergie électrique en fonction du le produit de la puissance électrique et du temps d'utilisation:

$$E = P \times \Delta t$$

Où E : représente l'énergie électrique (J), P : représente la puissance (W) et Δt : représente le temps (s). On a également:

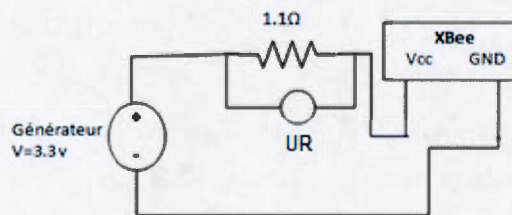
$$P = V \times I$$

Pour trouver la puissance, il faut déterminer le courant à partir de la loi d'Ohm :

$$UR = R \times I$$

$$\text{Donc } I = \frac{UR}{R}$$

Ce calcul se fait selon le schéma de montage ci-dessous:



$$P = V \times \frac{UR}{R}$$

$$P = 3.3 \times \frac{UR}{1.1}$$

On cherche alors la valeur de la tension UR pour pouvoir calculer l'énergie.

Nos résultats ont été obtenus à l'aide d'un oscilloscope de type Fluke 199C. Ce dernier affiche la valeur moyenne de la tension mesurée qui est de 0.33 (mV) en fonction du temps qui est égal à 200ms en cas de veille des nœuds Xbee. (FIG. 4.7)

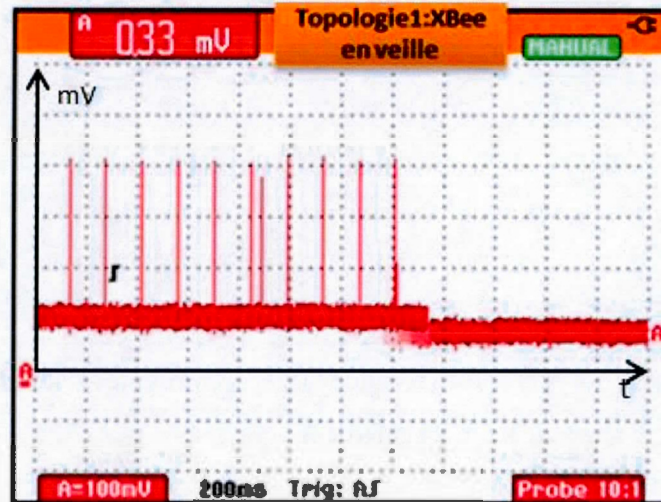


FIG. 4.8: Calcul de la tension des XBee en état de veille (topologie 1)

Un simple calcul donne:

$$P = 3.3 \times \frac{0.00033}{1.1} = 0.00099 \text{ w}$$

$$E = 0.00099 \times 0.2s \cong 0.0002J$$

Après l'envoi des données du nœud d'extrémité au nœud coordinateur, on remarque que la valeur de la tension visualisée augmente. Cela influence directement l'énergie consommée qui est de 0.0002 J en état de veille. Cependant, elle prend la valeur de 0.0025 J en état de marche soit 10 fois plus que quand les dispositifs s'endorment. (FIG. 4.8)

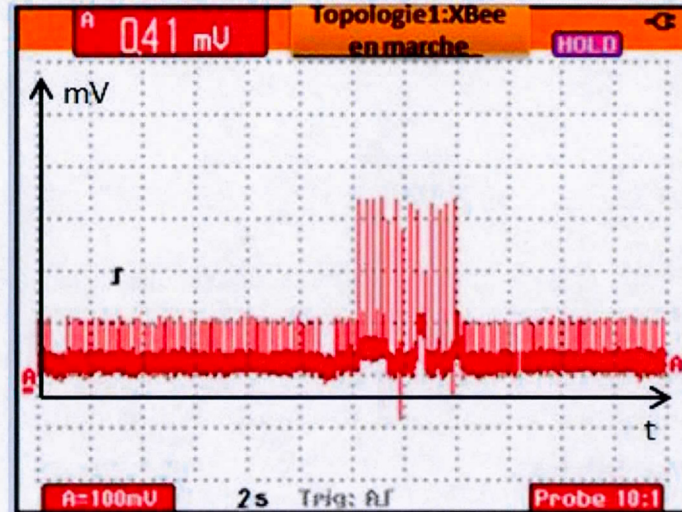


FIG. 4.8: Calcul de la tension des XBee en état de marche (topologie 1)

La puissance et l'énergie sont exprimées comme suit:

$$P = 3.3 \times \frac{0.00041}{1.1} = 0.00123 \text{ w}$$

$$E = 0.00123 \times 0.2s \cong 0.00025 \text{ J}$$

4.1.4 Deuxième topologie

Un deuxième nœud d'extrémité a été ajouté dans cette topologie telle que montrée sur la FIG. 4.9. Ce nœud reçoit des données captées par un capteur de force. Le nœud routeur reçoit plus de données par rapport à celui de la topologie 1; ce qui aura un impact sur le délai et l'énergie consommée.

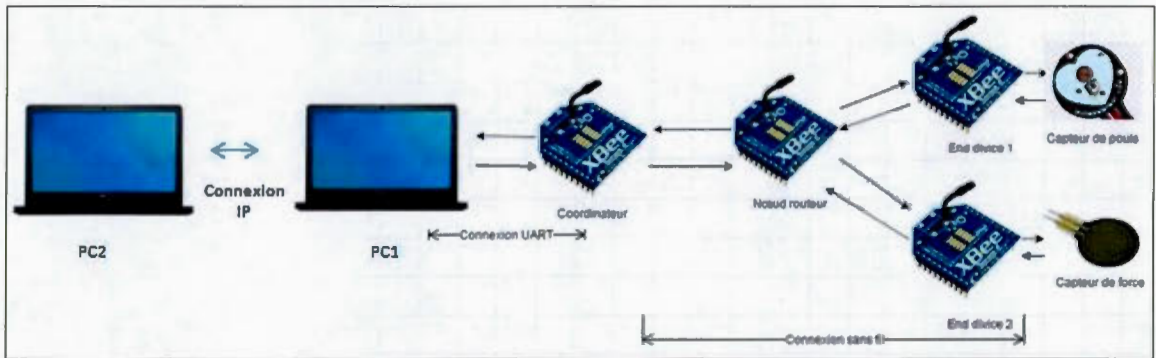


FIG. 4.9: Architecture du deuxième réseau ZigBee

Le nœud capteur qui possède l'adresse 00-13-A2-00-40-AF-BB-CF est configuré lui aussi en utilisant le logiciel X-CTU. La FIG. 4.10 ci-dessous nous permet de voir l'état du nouveau réseau.

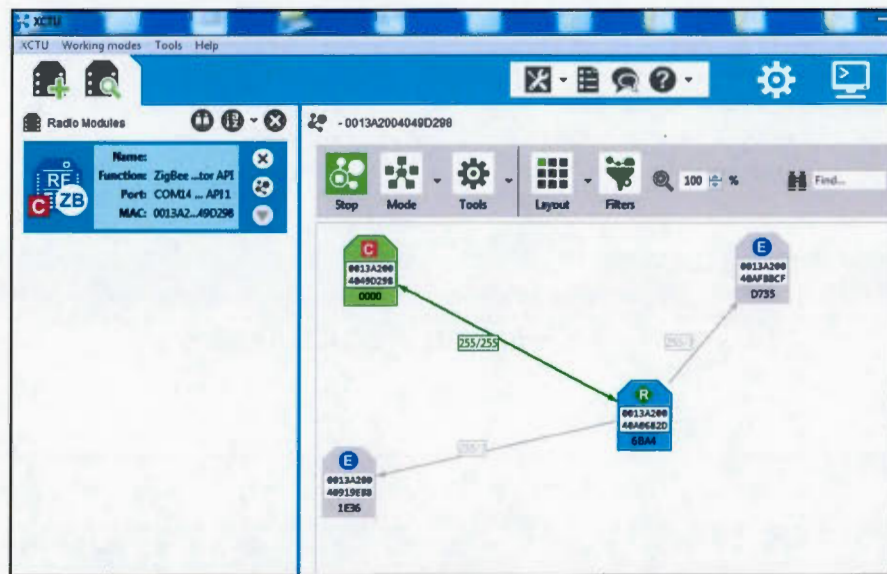


FIG. 4.10: Simulation du deuxième réseau avec X-CTU

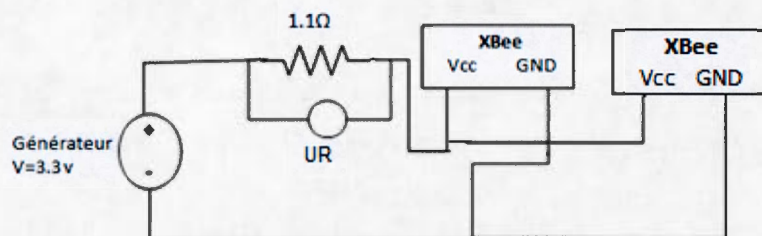
Les codes Java ont été adaptés à cette topologie afin qu'ils prennent en charge le deuxième nœud d'extrémité ajouté. L'estimation du délai de transmission de données est faite à trois reprises. Le fichier qui contient les données captées par les deux

capteurs sera téléchargé sur le PC2 à partir de son URL, sa taille est de 648 octets. Le TAB. 4.2 nous montre les résultats trouvés :

TAB. 4.2: Délai moyen de la topologie 2

	Experience1	Experience2	Experience3	Moyenne (ms)
Topologie 2	180,052	186,052	170,052	178,052

Le schéma de montage ci-dessous nous permet de mieux comprendre la manière dont on a branché les dispositifs pour arriver à visualiser la tension à l'aide du même oscilloscope (voir FIG. 4.11).



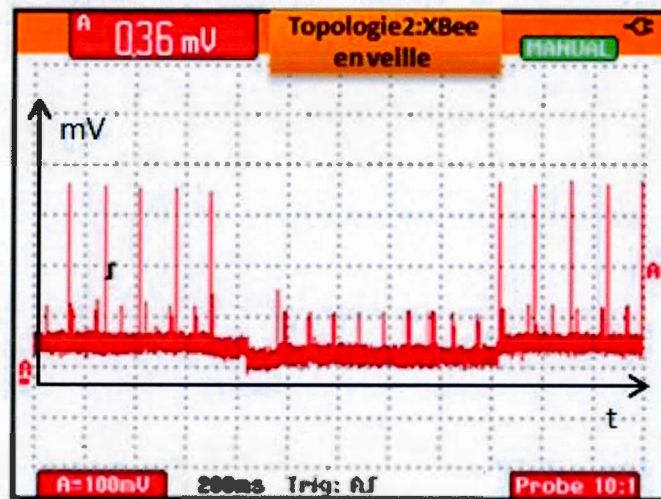


FIG. 4.11: Calcul de la tension des XBee en état de veille (topologie 2)

La puissance et l'énergie sont obtenues comme suit :

$$P = 3.3 \times \frac{0.00036}{1.1} = 0.00108 \text{ w}$$

$$E = 1.08 \times 0.2s \cong 0.00022 \text{ J}$$

On remarque que l'ajout d'un nœud capteur en état dormant influence légèrement sur la consommation d'énergie du système. On obtient 0.00022 J pour la deuxième topologie contre 0.0002 J pour la première.

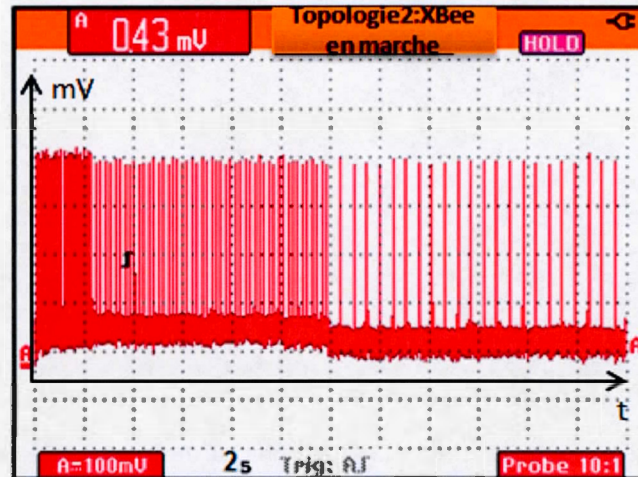


FIG. 4.12 : Calcul de la tension des XBee en état de marche (topologie 2)

La puissance et l'énergie sont obtenues comme suit :

$$P = 3.3 \times \frac{0.00043}{1.1} = 0.0013 \text{ w}$$

$$E = 0.0013 \times 0.2s \cong 0.00027 \text{ J}$$

La FIG. 4.12 permet de visualiser le comportement des nœuds XBee en état de marche pour la deuxième topologie. L'énergie consommée est de 0.00027 J.

4.1.5 Troisième topologie

Un troisième nœud capteur XBee a été ajouté au réseau (FIG. 4.14). Pour mieux comprendre l'influence de ce dispositif sur le délai de traitement des données et sur la consommation d'énergie, on a refait les mêmes expériences comme pour les autres topologies.

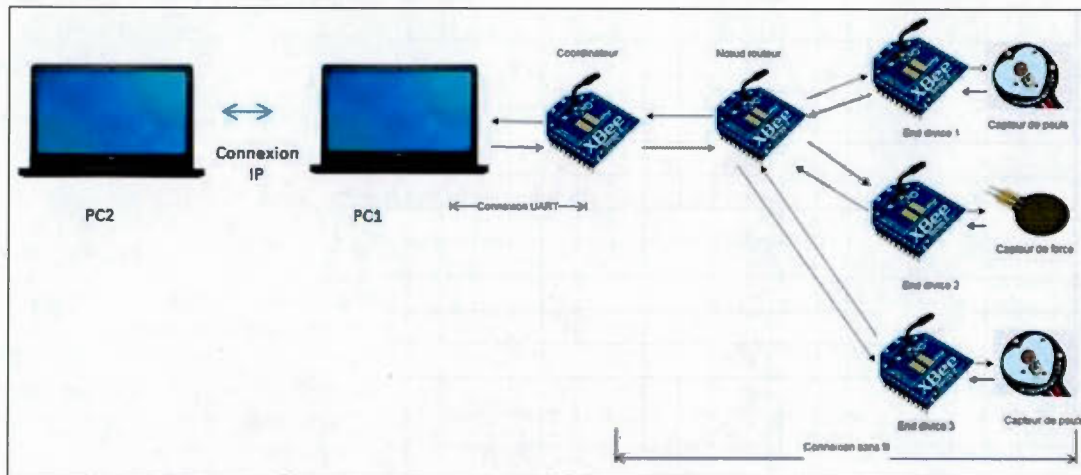


FIG. 4.13: Architecture du troisième réseau ZigBee

La FIG. 4.15 nous permet de vérifier l'état du nouveau réseau après la configuration du nœud d'extrémité dont l'adresse est 00-13-A2-00-40-91-9E-B8.

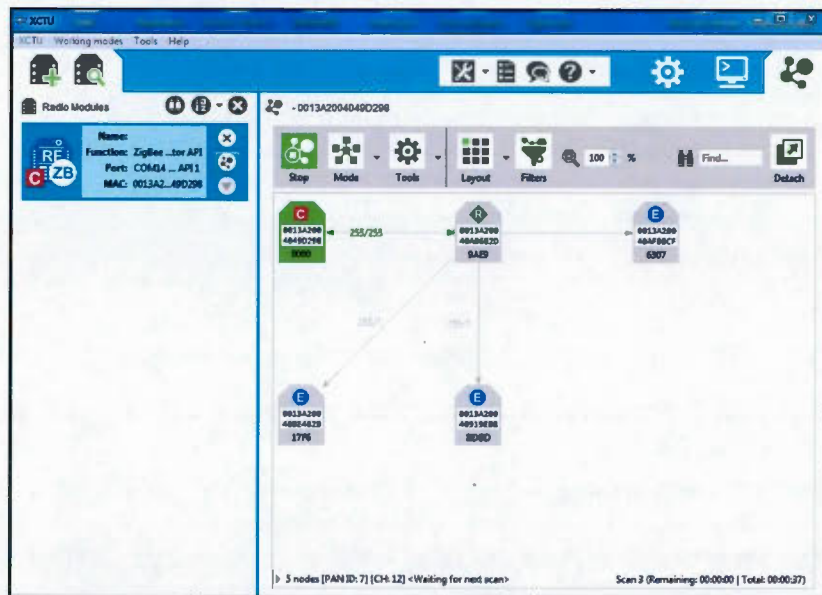


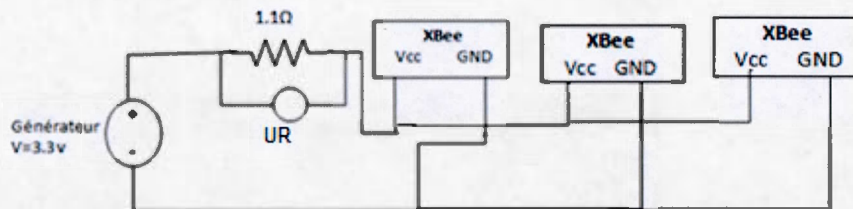
FIG. 4.14: Configuration du troisième réseau avec X-CTU

Le code Java a été modifié encore une fois pour prendre en charge le dispositif ajouté. Le TAB. 4.3 indique le délai moyen calculé. Le fichier des données captées envoyées au PC1, est de 901 octets et sera téléchargé sur le PC2 pendant 0.073ms.

TAB. 4.3: Délai moyen de la topologie 3

	Experience1	Experience2	Experience3	Moyenne (ms)
Topologie 3	196. 073	223. 073	219. 073	213.073

Le schéma de montage de ce réseau se compose de trois blocs XBee mise en série. Les résultats de la tension UR mesurée par le même outil nous permettent d'avoir une idée sur la consommation d'énergie du réseau.



En état de veille, la tension UR est égale à 0.38 (mV) comme montré sur la FIG. 4.16. Le calcul de l'énergie est fait comme suit :

$$P = 3.3 \times \frac{0.00038}{1.1} = 0.00114 \text{ w}$$

$$E = 1.14 \times 0.2s \cong 0.00023J$$

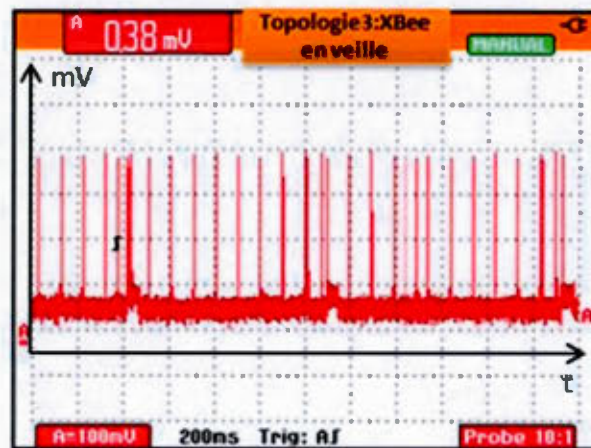


FIG. 4.15: Calcul de la tension des XBee en état de veille (topologie 3)

La tension UR mesurée est de 0.51 mV quand les nœuds XBee sont en veille. Le calcul de l'énergie consommée sera de 0.0036 J pour la troisième topologie. (FIG. 4.17).

$$P = 3.3 \times \frac{0.00051}{1.1} = 0.00153 \text{ w}$$

$$E = 0.00153 \times 0.2s \cong 0.000306J$$



FIG. 4.16: Calcul de la tension des XBee en état de marche (topologie 3)

TAB. 4.4: Comparaison des résultats

	Délai de transmission (ms)	Consommation d'énergie (J)
ZigBee 1	79.03	0.00013
ZigBee 2	178.052	0.00027
ZigBee 3	213.073	0.00031

Les résultats des trois topologies de réseau ZigBee effectuées ont été résumés dans le TAB. 4.4 afin de les comparer.

Nous remarquons que le délai de transmission des données augmente de quelques dizaines de ms à chaque fois qu'un nouveau dispositif s'ajoute au réseau. Les nœuds XBee sont conçus pour fonctionner longtemps. La consommation d'énergie des trois topologies de réseau est faible comme nous montrent les résultats.

Il y a une forte probabilité que le délai et la consommation d'énergie augmenteront plus si on se retrouve face à un réseau avec un grand nombre de nœuds capteurs. De nouveaux mécanismes doivent s'appliquer dans ce cas.

Les nœuds capteurs sont déployés généralement dans des environnements hostiles. La mise en œuvre et la défaillance des nœuds sont donc compliquées et très probables.

4.2 Réalisation logicielle du réseau ZigBee IP

4.2.1 Première simulation

Nous avons opté pour l'environnement de développement Contiki afin de réaliser les simulations du réseau ZigBee IP. Il est sous forme d'une machine virtuelle 'Vmware' qui contient le code source et toutes ses fonctionnalités. Il contient notamment un

simulateur/émulateur appelé Cooja qui permet de simuler les connexions réseaux de capteurs appelés aussi 'motes' et d'interagir avec les capteurs et afficher les événements au fil du temps. Les motes peuvent être émulés au niveau hardware ou à un niveau moins détaillé comme dans notre cas.

Plusieurs types de capteurs sont supportés par Cooja tels que *Z1*, *EXP5438*, *Sicaz*, *sky*, *Jcreate*, etc. Nous avons choisi la carte de capteurs sans fil 'Tmote sky' intégrée dans Contiki, compatible avec la norme 802.15.4. Les types de capteurs à faible puissance utilisés comprennent généralement de capteurs de mesure de l'humidité, de la température et de la lumière.

Cooja dispose de plusieurs types de medias sans fil. Nous travaillons sur l'UDGM (Unit Disk Graph Medium) offert par défaut dans lequel les interférences sont prises en considération et que le taux des paquets transmit et reçut est faite avec succès.

La première simulation que nous avons effectuée se compose de cinq nœuds (FIG. 4.18). Le premier nœud est un routeur de frontière 6LoWPAN avec une compression d'en-tête IPv6 à 6LoWPAN. Ce nœud est connecté à un ordinateur fonctionnant sous Linux. Les autres nœuds se considèrent comme des dispositifs de 6LoWPAN et signalent les données envoyées par les capteurs.

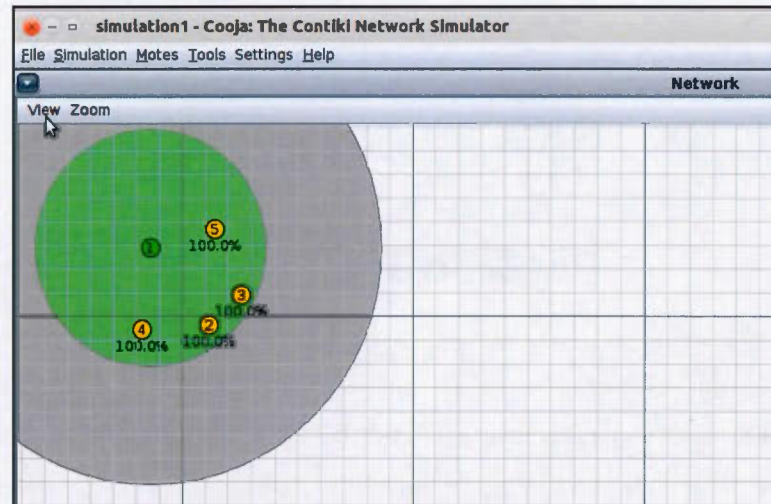


FIG. 4.17: Simulation du premier exemple de réseau ZigBee IP

Cette application est déployée avec l'adressage IPv6 pour tous les nœuds. Pour être en mesure de lire les données à partir d'un serveur web, il faut connecter le routeur de frontière avec une commande Unix : `$ make connect-router-cooja` (FIG. 4.18)

```
user@instant-contiki:~/contiki-2.7/examples/lpvr/rpl-border-routers$ make connect-router-cooja
TARGET not defined, using target 'native'
sudo ./../tools/tunslip -a 127.0.0.1 aaaa::1/64
[sudo] password for user:
slip connected to "127.0.0.1:60001"
opened tun device "/dev/tun0"
ifconfig tun0 inet 'hostname' up
ifconfig tun0 add aaaa::1/64
ifconfig tun0 add fe80::0:0:0:1/64
ifconfig tun0

tun0    Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
        inet addr:127.0.0.1 P t P:127.0.0.1 Mask:255.255.255.255
        inet6 addr: fe80::1/64 Scope:Link
        inet6 addr: aaaa::1/64 Scope:Global
        UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:500
        RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

line started with address 0.1B.110.1.0.1.1.1
MAC 00:12:74:01:00:01:01:01 Contiki-2.6-900-ga6227e1 started. Node id is set to 1.
CSMA ContikiMAC, channel check rate 8 Hz, radio channel 26, CCA threshold -45
tentative link local IPv6 address fe80:0000:0000:0000:0212:7401:0001:0101
Starting "Border router process" "Web server"
*** Address:aaaa::1 => aaaa:0000:0000:0000
Got configuration message of type P
getting prefix aaaa::
server IPv6 addresses:
aaaa::212:7401:1:101
fe80::212:7401:1:101
```

FIG. 4.18: Connexion du nœud routeur

Le lien entre les nœuds capteurs s'établit une fois que le pont serveur est activé comme montré sur la FIG. 4.19

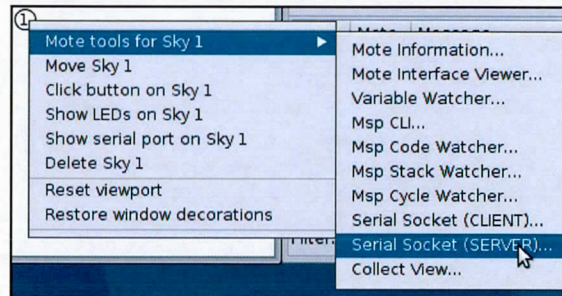


FIG. 4.19: Activation du pont du nœud serveur

Le protocole de découverte de voisin (NDP) se trouve au niveau de la couche liaison. Il est utilisé par l'IPv6 afin de découvrir les autres nœuds sur le réseau ayant la même liaison radio. On est alors en mesure d'interroger chaque nœud avec son adresse IPv6 pour pouvoir capter les données collectées.

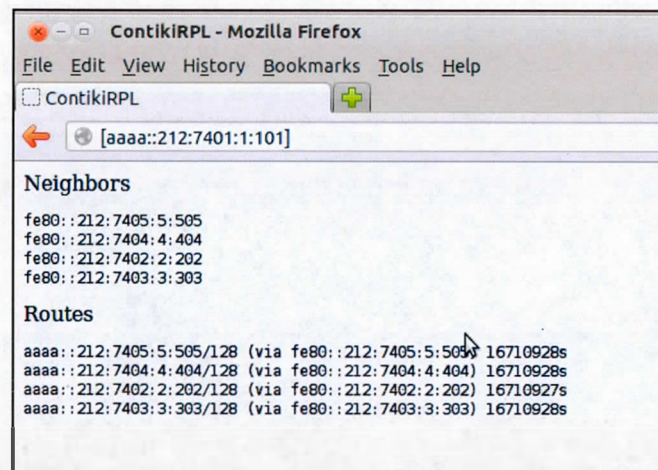


FIG. 4.20: Découverte de voisin à partir du nœud serveur

Chaque nœud capteur dans le réseau fournit les détails sur les données envoyées par le capteur. Dans notre cas, on a interrogé quatre nœuds à travers leur adresse IPv6 telle que montrée sur la FIG. 4.21 ci-dessous :

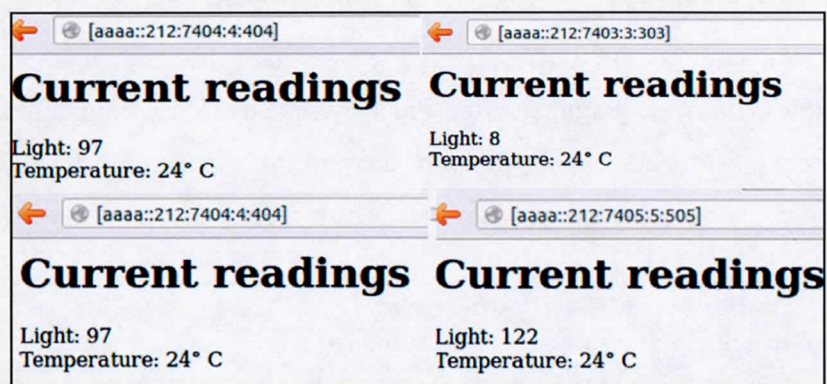


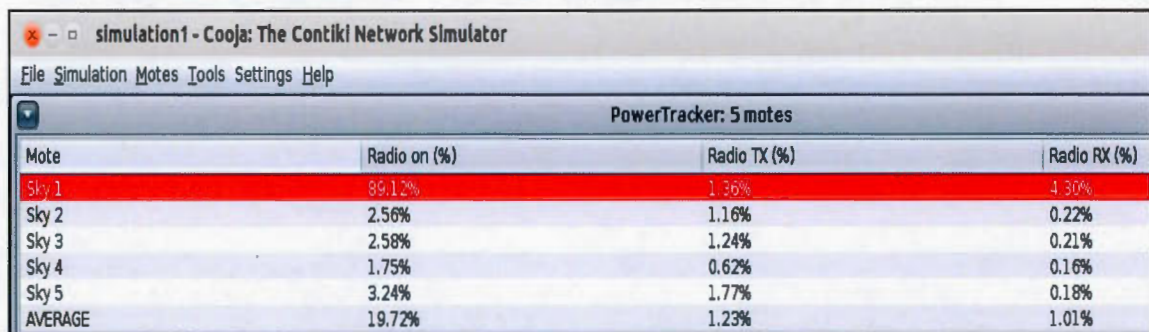
FIG. 4.21: Valeur des capteurs des quatre nœuds du réseau

Les nœuds capteurs du réseau ont été testés trois fois pour être en mesure de capter différentes valeurs comme montrées sur le (TAB. 4.5).

TAB. 4.5: Données captées par les nœuds du réseau

	Test 1		Test 2		Test 3	
Lumière/Température(C)	L	T	L	T	L	T
Nœud capteur 2	257	24	271	24	5	23
Nœuds capteur 3	211	22	168	24	185	25
Nœud capteur 4	182	24	225	24	108	24
Nœud capteur 5	172	21	220	19	166	24

L'outil 'PowerTracker' de Cooja, nous permet de visualiser directement l'utilisation du canal radio pour chacun des nœuds capteurs du réseau. Le paramètre 'Radio On' exprime en pourcentage l'utilisation du transceiver sans fil de chaque nœud, y compris le temps d'émission, de réception ainsi que pour l'état du repos. De même, les paramètres 'Radio TX' et 'Radio RX' expriment le temps de transmission et de réception respectivement pour chaque nœud du réseau. Le premier nœud 'Sky 1' agit comme coordinateur du réseau. Il envoie des messages de diffusion et reçoit les données de tous les autres nœuds en utilisant les mécanismes 6LoWPAN de compression et de fragmentation. Par conséquent, sa consommation d'énergie reste beaucoup plus élevée par rapport aux autres nœuds. La FIG. 4.22 montre la consommation d'énergie pour tous les nœuds du réseau.



The screenshot shows a window titled 'simulation1 - Cooja: The Contiki Network Simulator'. Inside, there's a menu bar with 'File', 'Simulation', 'Motes', 'Tools', 'Settings', and 'Help'. Below the menu is a tab labeled 'PowerTracker: 5 motes'. The main area contains a table with the following data:

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	89.12%	1.36%	4.30%
Sky 2	2.56%	1.16%	0.22%
Sky 3	2.58%	1.24%	0.21%
Sky 4	1.75%	0.62%	0.16%
Sky 5	3.24%	1.77%	0.18%
AVERAGE	19.72%	1.23%	1.01%

FIG. 4.22: Consommation d'énergie des nœuds du premier réseau.

La consommation d'énergie a un lien direct avec la puissance. Dans notre cas la puissance dépend d'un paramètre de temps T . Ce dernier est sous forme de pourcentage lié à l'utilisation de la radio en émission et en réception. La consommation d'énergie reste proportionnelle à l'utilisation de la radio. Les équations suivantes expriment l'utilisation de la puissance et de l'énergie dans le cas de ce réseau créée par Contiki:

$$Puissance = P^T(mW) = T \times 20mA \times 3V / 4096$$

$$Énergie = Puissance \times \text{temps de l'exécution (seconds)}$$

T : temps de fin d'utilisation de la radio - temps de début d'utilisation de la radio. Le courant électrique reste constant (20mA). La tension opérationnelle des nœuds TmoteSky est de 3V. Le nombre de ticks par seconde est de 4096. Le temps de l'exécution est indiqué sur le l'outil de contrôle de simulateur de Cooja, mais pour celui de la 'radio On' on peut le récupérer automatiquement à l'aide de la commande unix : *gedit timer radio_on*

```
[java] AVERAGE radio_on 127666 us 0.008287315178950694 %
```

L'énergie totale consommée par cette topologie de réseau peut être calculée comme suit :

$$P = \frac{1.23 \times 20 \times 3}{4096} = 0.00028 \text{ w}$$

$$E = 0.00028 \times 0.127 = 3,36 \times 10^{-4} \text{ J}$$

L'énergie consommée quand la radio est en train d'émettre peut être calculé de la façon suivante :

$$P = \frac{19.72 \times 20 \times 3}{4096} = 0.000018 \text{ w}$$

$$E = 0.00028 \times 0.127 = 2,28 \times 10^{-5} \text{ J}$$

4.2.2 Deuxième simulation

La topologie créée dans la deuxième simulation comporte sept nœuds capteurs. Contrairement à la première simulation, les nœuds 2, 5 et 7 n'appartiennent pas au

champ de transmission du nœud coordinateur. Le grand cercle montré sur la FIG. 4.23 représente le cercle d'interférences. Un nœud dans cette zone n'est pas en mesure de recevoir les données correctement, ni de communiquer avec les autres quand le nœud coordinateur est en train d'émettre ou de recevoir.

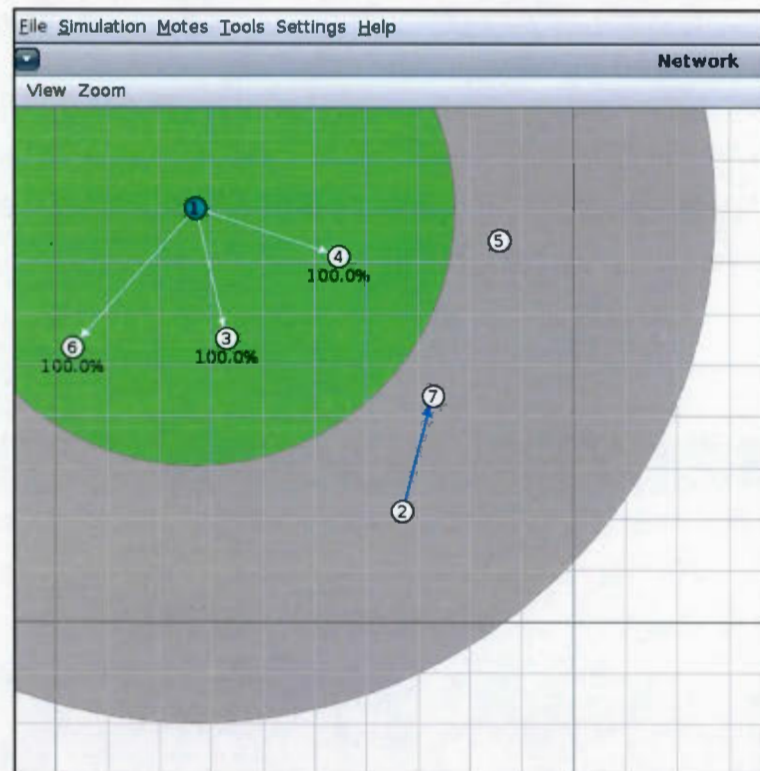


FIG. 4.23: Simulation du deuxième exemple du réseau ZigBee IP

Les nœuds qui ne se trouvent pas dans le champ de transmission du nœud coordinateur ne se considèrent pas comme des nœuds voisins (FIG. 4.24). Cependant toutes les routes créées sur le réseau sont contrôlées par le nœud 1. Tous les nœuds peuvent être interrogés en utilisant leur adresse IPv6.

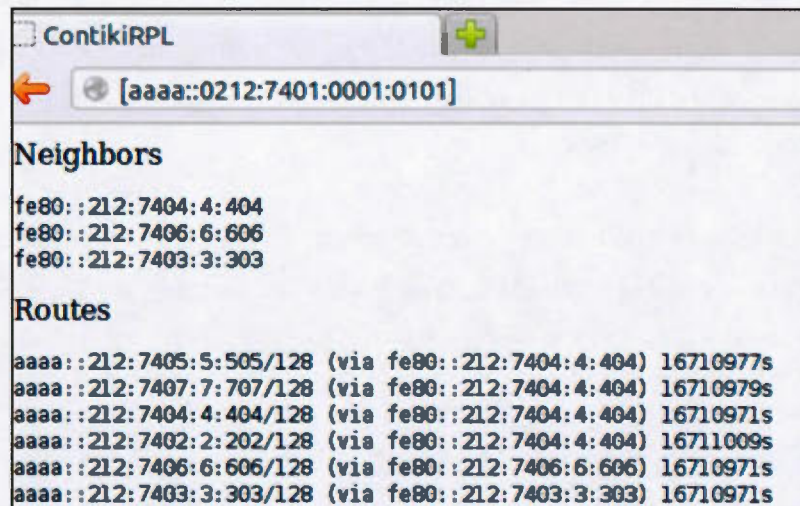


FIG. 4.24: Découverte de voisin pour la deuxième topologie

Le nœud serveur utilise plus de ressources radio par rapport au nœud serveur de la première topologie. Il consomme 93.93% de ces ressources (FIG. 4.25), cela est dû au nombre de nœuds qu'il gère et supervise. La consommation moyenne de tout le système (1 nœud serveur et 7 nœuds capteur) est de 15.27%. On remarque que la consommation moyenne d'énergie a diminué par rapport à la première topologie qui a consommé 19.72%. Ce qui veut dire que le nombre de nœuds n'influence pas sur la consommation d'énergie dans les réseaux ZigBee IP.

PowerTracker: 7 motes			
Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	93.93%	1.13%	2.78%
Sky 2	2.61%	1.23%	0.18%
Sky 3	1.99%	0.68%	0.28%
Sky 4	2.28%	0.74%	0.33%
Sky 5	2.57%	1.17%	0.23%
Sky 6	2.69%	1.31%	0.19%
Sky 7	2.00%	0.83%	0.16%
AVERAGE	15.27%	1.01%	0.59%

FIG. 4.25: Consommation d'énergie du deuxième réseau

CoAP utilise le chemin `/.Well-know/core` pour la découverte des ressources. Ce chemin définit les attributs suivants : 'rt' est le type du contenu, 'ct' est la taille des données, 'sz' est la taille maximale supportée. Pour visualiser ces attributs, il faut se mettre en mode privilège 'core' et taper la commande GET.

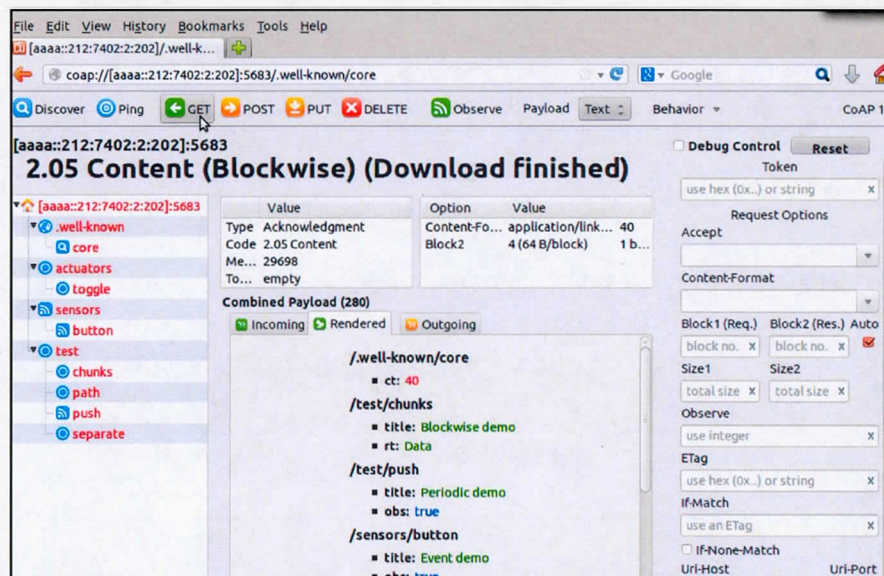


FIG. 4.27: Ressources offerte par CoAP

Pour pouvoir interroger un capteur, il faut tout d'abord découvrir les ressources existantes, se mettre en mode privilège et taper la commande 'observe'. La FIG. 4.28, permet de visualiser les données captées par le deuxième nœud capteur :

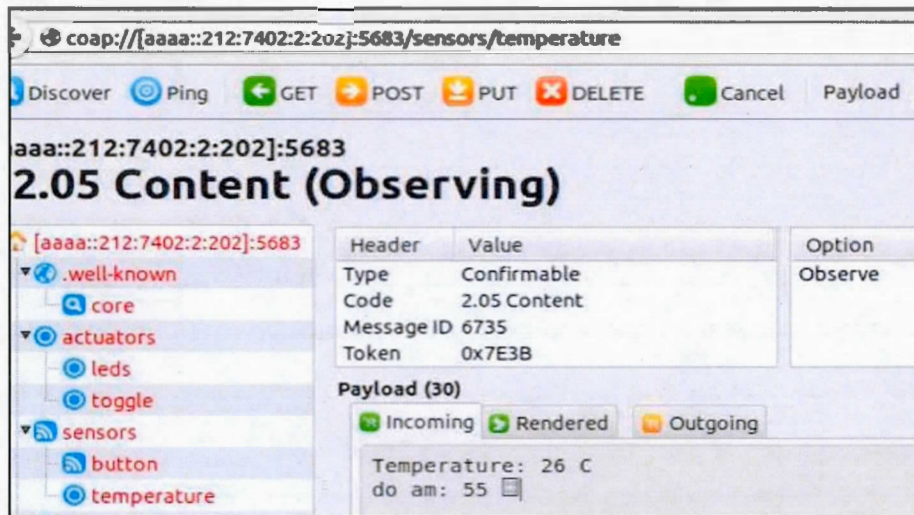


FIG. 4.28: Interrogation du nœud capteur 2

Le test a été effectué 3 fois, le TAB. 4.6 nous montre les résultats obtenus:

TAB. 4.6: Données captées par le nœud 2

Température	Lumière
26	55
26	109
25	206

CoAP nous permet aussi de passer d'un état à un autre par des simples commandes. L'action 'toggle' nous permet de basculer l'état des LED intégrées dans l'environnement de simulation. Comme nous le montre la FIG. 4.29, en tapant '1' suivi de la commande POST on sera capable d'allumer la LED rouge des nœuds capteur 3.

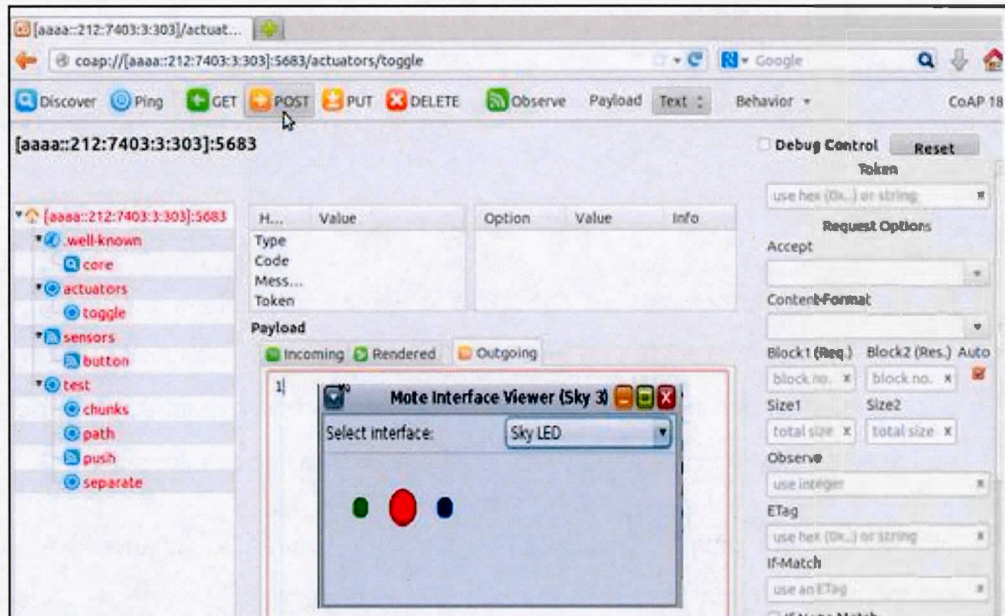


FIG. 4.29: Changement d'état de la LED du nœud capteur 3

4.3 Comparaison des résultats

Pour mieux comprendre l'intérêt des tests et des simulations effectués, on va mettre en comparaison les résultats de la troisième topologie du réseau ZigBee-Pro obtenus avec les dispositifs XBee avec les résultats de la première topologie du ZigBee IP développé sur Contiki. Le choix de ces deux topologies a été fait en se basant sur le nombre de nœuds utilisés qui de cinq nœuds pour chacune d'elles.

Le TAB. 4.7 suivant nous résume ces résultats :

TAB. 4.7: Comparaison des résultats entre ZigBee-Pro et ZigBee IP

	ZigBee-Pro	ZigBee IP
Délai	<i>213.073ms</i>	<i>127 ms</i>
Énergie	<i>0.00031 J</i>	$2,28 \times 10^{-5} \text{ J}$

Les deux environnements de travail se diffèrent complètement. Pour ZigBee-Pro, nous avons émulé avec du vrai matériel. On constate qu'il prend plus de temps et plus d'énergie, mais permet une inspection précise du comportement du système, détaillée vu les configurations des paramètres ainsi que le contrôle des données.

ZigBee IP a été simulé virtuellement à l'aide de Contiki. L'échange des données se fait plus rapidement et la consommation d'énergie a été optimisée. Cela est dû à l'utilisation des mécanismes avancés comme l'IPv6, 6LoWPAN. En plus, l'outil de développement Contiki assure une grande flexibilité du réseau, gère le contrôle de la plateforme avec un moindre coût.

CONCLUSION

Dans ce mémoire, plusieurs standards sans fil de faible puissance utilisant la norme IEEE 802.15.4 comme support de base ont été présentés. Dans un premier temps, nous avons donné une description approfondie des principaux réseaux utilisés.

ZigBee, un réseau robuste, attractif et à faible coût. Sa consommation électrique est très faible et a la possibilité de déployer de réseaux maillés à très grande échelle dont la limite théorique atteint 65536 nœuds du réseau. Il est destiné au contrôle, à la surveillance et à la gestion de commandes à distance. ZigBee peut être appliqué dans différents domaines comme la domotique automatisée et beaucoup d'autres domaines d'actualité. Ces caractéristiques lui ont permis de s'intégrer dans la nouvelle révolution d'Internet des Objets (IoT) à travers ces deux versions ZigBee-Pro et ZigBee IP.

Nous avons présenté les spécifications de chaque version. ZigBee Pro est une extension directe de ZigBee d'origine. Ces mêmes dispositifs ont été améliorés en termes de puissance d'émission et de portée en plus de l'ajout de la passerelle qui lui a permis d'être relié à Internet. Zigbee IP est une version très améliorée de ZigBee, considérée comme le premier standard ouvert compatible avec le protocole IPv6. Elle assure l'établissement de la connexion d'une manière transparente avec des dispositifs à faible puissance et à faible cout. La couche d'adaptation 6LoWPAN intégrée dans ZigBee IP garantit l'interopérabilité entre la couche liaison définie par la norme 802.15.4 et la couche réseau qui supporte l'IPv6, cela est pour lier n'importe quel appareil à l'internet.

Dans le cadre de la comparaison entre les deux versions ZigBee Pro et ZigBee IP, nous avons effectué des réalisations matérielles et logicielles pour être en mesure de

bien déterminer les différences entre les deux en termes de délai et de consommation d'énergie.

Les réalisations matérielles ont été faites à l'aide des dispositifs XBee pro de ZigBee. Les topologies utilisées et testées ont montré que le délai de transmission et la consommation d'énergie augmentent si le réseau s'élargit.

La réalisation logicielle est sous forme d'une simulation du réseau ZigBee IP. Elle a été développée sur le système d'exploitation Contiki. Ce dernier est un nouvel outil dédié à l'internet des objets. Les simulations effectuées ont montré que le nombre de nœuds n'influence pas sur la consommation d'énergie. Des améliorations sont liées à l'utilisation des mécanismes avancés de 6LoWPAN permettent une meilleure exploitation d'énergie et de délai de traitement de données. Une adresse IPv6 de 128 bits peut être compressée jusqu'à 16 bits en utilisant 6LoWPAN. L'avantage des dispositifs 6LoWPAN est qu'ils peuvent être abordés de partout dans le monde s'ils sont bien configurés.

Travaux futurs :

Le sujet que nous avons traité dans ce mémoire est en plein essor ces derniers temps. Nous présentons ici quelques idées pour les travaux futurs :

- Améliorer le débit du réseau ZigBee en ajoutant de nouveaux dispositifs.
- La comparaison des deux versions de ZigBee peut se faire en se basant sur d'autres critères.
- L'ajout de nouveaux mécanismes de routage pour le réseau ZigBee dans le cas d'un réseau encombré.
- L'utilisation des dispositifs matériels de 6LoWPAN pour une exploitation précise du comportement des systèmes.

BIBLIOGRAPHIE

- Alliance, Z. (2008). Zigbee Specification. *Zigbee Alliance Website*, 1–604.
http://doi.org/http://people.ece.cornell.edu/land/courses/ece4760/FinalProjects/s2011/kjb79_ajm232/pmeter/ZigBee%20Specification.pdf
- Baker, N. (2005). ZigBee and Bluetooth: strengths and weaknesses for industrial applications. *Computing and Control Engineering*, 16(2), 20–25.
<http://doi.org/10.1049/cce:20050204>
- Bechkit, W., Bouabdallah, A., & Challal, Y. (2011). Un prototype de réseaux de capteurs sans fil pour l'agriculture et le contrôle de l'environnement. *CFIP 2011 - Colloque Francophone Sur L'Ingénierie Des Protocoles*.
- Bormann, C., Castellani, A. P., & Shelby, Z. (2012). CoAP: An application protocol for billions of tiny internet nodes. *IEEE Internet Computing*, 16(2), 62–67.
<http://doi.org/10.1109/MIC.2012.29>
- Boushaba, A., Oumsis, M., & Benabbou, R. (2010). Evaluation des performances des protocoles de routage Ad hoc. *Jdir*.
- Chehaider, M., Roussel, K., & Song, Q. (2013). Intéropérabilité des réseaux de capteurs hétérogènes dans un appartement intelligent. *Ubimob 2013*, 5–6.
- F. Shanin. (2008). *ZigBee and IEEE 802.15.4 Protocol Layers. ZigBee Wireless Networks and Transceiver*.
- Franceschinis, M., Pastrone, C., Spirito, M. A., & Borean, C. (2013). On the performance of ZigBee Pro and ZigBee IP in IEEE 802.15.4 networks.

International Conference on Wireless and Mobile Computing, Networking and Communications, 83–88. <http://doi.org/10.1109/WiMOB.2013.6673344>

Francomme, J. (n.d.). Propositions pour un protocole d'exterministe de contr^es et de routage avec 'economie d' 'energie dans les r 'eseaux ZigBee Systèmes Communicants Sans Fil Maître ès Sciences Université Paris XII « Propositions pour un protocole déterministe de cont.

Hui, J. W., & Culler, D. E. (2010). IPv6 in Low-Power Wireless Networks. *Proceedings of the IEEE*, 98(11), 1865–1878. <http://doi.org/10.1109/JPROC.2010.2065791>

International, D. (2009). XBee ® /XBee-PRO ® RF Modules. *Product Manual v1.xEx-802.15.4 Protocol*, 1–69. Retrieved from <http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/xbee-series1-module#specs>

Ismail, N. H. A., Hassan, R., & Ghazali, K. W. M. (2012). A study on protocol stack in 6lowpan model. *Journal of Theoretical and Applied Information Technology*, 41(2), 220–229.

Kanthe, A. M., Simunic, D., & Prasad, R. (2012). Comparison of AODV and DSR on-demand routing protocols in mobile ad hoc networks. *1st International Conference on Emerging Technology Trends in Electronics, Communication and Networking (ET2ECN)*, 1–5. <http://doi.org/10.1109/ET2ECN.2012.6470118>

Keng Ee, G., Kyun Ng, C., Kamariah Noordin, N., & Mohd Ali, B. (2010). A Review of 6LoWPAN Routing Protocols. *Proceedings of the Asia-Pacific Advanced Network*, 30, 71–81. <http://doi.org/10.7125/APAN.30.11>

Kovatsch, M., Weiss, M., & Guinard, D. (2010). Embedding internet technology for

- home automation. *IEEE 15th Conference on Emerging Technologies & Factory Automation (ETFA 2010)*, 1–8. <http://doi.org/10.1109/ETFA.2010.5641208>
- Kumar, A., & Singh, C. (2011). Performance evaluation of routing protocols for packet drop statistics for Meshed routing in IEEE 802 . 15 . 4 based WSNs, *7109*, 50–56.
- Lee, J. S., Su, Y. W., & Shen, C. C. (2007). A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. *IECON Proceedings (Industrial Electronics Conference)*, 46–51. <http://doi.org/10.1109/IECON.2007.4460126>
- Maurizio Pellegrini, R., Persia, S., Volponi, D., & Marcone, G. (2010). ZigBee Sensor Network propagation analysis for health-care application. In *2010 Fifth International Conference on Broadband and Biomedical Communications* (pp. 1–6). IEEE. <http://doi.org/10.1109/IB2COM.2010.5723635>
- MaxStream. (2005). XBee TM / XBee-PRO TM OEM RF Modules. *MaxStream*, (801), 1–33.
- Mazzer, Y., & Tourancheau, B. (2009). Comparisons of 6LoWPAN implementations on wireless sensor networks. *Proceedings - 2009 3rd International Conference on Sensor Technologies and Applications, SENSORCOMM 2009*, 689–692. <http://doi.org/10.1109/SENSORCOMM.2009.111>
- Mulligan, G. (2007). The 6LoWPAN architecture. *Proceedings of the 4th Workshop on Embedded Networked Sensors - EmNets '07*, 78. <http://doi.org/10.1145/1278972.1278992>
- Nemri, M. (2015). Demain, l'Internet des objets. *Note D'analyse - France.gouv.*
- Österlind, F., Eriksson, J., & Dunkels, A. (2010). Cooja TimeLine: a power visualizer for sensor network simulation. *Proceedings of the 8th ACM Conference on*

- Embedded Networked Sensor Systems (SenSys 2010)*, 385–386.
<http://doi.org/10.1145/1869983.1870035>
- Pediredla, B., Wang, K. I. K., Salcic, Z., & Ivoghlian, A. (2013). A 6LoWPAN implementation for memory constrained and power efficient wireless sensor nodes. *IECON Proceedings (Industrial Electronics Conference)*, 4432–4437.
<http://doi.org/10.1109/IECON.2013.6699849>
- Phan, D., & Contr , D. P. (2015). Contr le de la puissance pour les r seaux sans fil Universit  de Poitiers.
- Pinedo-Frausto, E. D., & Garcia-Macias, J. A. (2008). An experimental analysis of zigbee networks. *Proceedings - Conference on Local Computer Networks, LCN*, 723–729. <http://doi.org/10.1109/LCN.2008.4664272>
- Sehgal, A. (2013). Using the Contiki Cooja Simulator, 1–7.
- Toscano, E., & Lo Bello, L. (2012). Comparative assessments of IEEE 802.15.4/ZigBee and 6LoWPAN for low-power industrial WSNs in realistic scenarios. In *2012 9th IEEE International Workshop on Factory Communication Systems* (pp. 115–124). IEEE. <http://doi.org/10.1109/WFCS.2012.6242553>
- Val, T., Campo, E., & Van Den Bossche, A. (2008). Technologie ZigBee/802.15.4 Protocoles, topologies et domaines d'application, 1–8. Retrieved from *Reseaux Informatiques/RESEAUX LOCAUX ET RESEAUX SANS FIL/te7508 Technologie ZigBee-802.15.4 - Protocoles, topologies .pdf*
- Zou, Z., Li, K. J., Li, R., & Wu, S. (2011). Smart home system based on IPV6 and ZIGBEE technology. *Procedia Engineering*, 15, 1529–1533.
<http://doi.org/10.1016/j.proeng.2011.08.284>